Research Topics

2022 Applied mathematics and statistics – editor's pick

Edited by
Charles K. Chui, Eric Chung, Jianfeng Cai, Raluca Eftimie, Hong-Kun Xu, Daniel Potts, Young Shin Aaron Kim and Axel Hutt

Published in
Frontiers in Applied Mathematics and Statistics

ISSN 1664-8714
ISBN 978-2-83252-006-2
DOI 10.3389/978-2-83252-006-2

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

2022 Applied mathematics and statistics – editor's pick

Topic editors

Charles K. Chui — Stanford University, United States
Eric Chung — The Chinese University of Hong Kong, China
Jianfeng Cai — Hong Kong University of Science and Technology, SAR China
Raluca Eftimie — University of Franche-Comté, France
Hong-Kun Xu — Hangzhou Dianzi University, China
Daniel Potts — Chemnitz University of Technology, Germany
Young Shin Aaron Kim — Stony Brook University, United States
Axel Hutt — Inria Nancy - Grand-Est research centre, France

Citation

Chui, C. K., Chung, E., Cai, J., Eftimie, R., Xu, H.-K., Potts, D., Kim, Y. S. A., Hutt, A., eds. (2023). *2022 Applied mathematics and statistics – editor's pick*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-83252-006-2

Table of contents

frontiers
in Applied Mathematics and Statistics

ORIGINAL RESEARCH
published: 13 January 2022
doi: 10.3389/fams.2021.779841

A Statistically and Numerically Efficient Independence Test Based on Random Projections and Distance Covariance

*Cheng Huang and Xiaoming Huo**

Georgia Institute of Technology, Atlanta, GA, United States

Testing for independence plays a fundamental role in many statistical techniques. Among the nonparametric approaches, the distance-based methods (such as the distance correlation-based hypotheses testing for independence) have many advantages, compared with many other alternatives. A known limitation of the distance-based method is that its computational complexity can be high. In general, when the sample size is n, the order of computational complexity of a distance-based method, which typically requires computing of all pairwise distances, can be $O(n^2)$. Recent advances have discovered that in the *univariate* cases, a fast method with $O(n \log n)$ computational complexity and $O(n)$ memory requirement exists. In this paper, we introduce a test of independence method based on random projection and distance correlation, which achieves nearly the same power as the state-of-the-art distance-based approach, works in the *multivariate* cases, and enjoys the $O(nK \log n)$ computational complexity and $O(\max\{n, K\})$ memory requirement, where K is the number of random projections. Note that saving is achieved when $K < n/\log n$. We name our method a Randomly Projected Distance Covariance (RPDC). The statistical theoretical analysis takes advantage of some techniques on the random projection which are rooted in contemporary machine learning. Numerical experiments demonstrate the efficiency of the proposed method, relative to numerous competitors.

Keywords: independence test, distance covariance, random projection, hypotheses test, multivariate hypothesis test

OPEN ACCESS

Edited by:
David Banks,
Duke University, United States

Reviewed by:
Dominic Edelmann,
German Cancer Research Center, Germany
Gabor J. Szekely,
National Science Foundation (NSF), United States

****Correspondence:***
Xiaoming Huo
huo@gatech.edu

Specialty section:
This article was submitted to Statistics, a section of the journal Frontiers in Applied Mathematics and Statistics

Received: *19 September 2021*
Accepted: *16 November 2021*
Published: *13 January 2022*

Citation:
Huang C and Huo X (2022) A Statistically and Numerically Efficient Independence Test Based on Random Projections and Distance Covariance. Front. Appl. Math. Stat. 7:779841. doi: 10.3389/fams.2021.779841

1 INTRODUCTION

Test of independence is a fundamental problem in statistics, with many existing work including the maximal information coefficient (MIC) [1], the copula based measures [2,3], the kernel based criterion [4] and the distance correlation [5,6], which motivated our current work. Note that the above works as well as ours focus on the testing for independence, which can be formulated as statistical hypotheses testing problems. On the other hand, interesting developments (e.g., [7]) aim at a more general framework for interpretable statistical dependence, which is not the goal of this paper.

Distance correlation proposed by [6] is an important method in the test of independence. The direct implementation of distance correlation takes $O(n^2)$ time, where n is the sample size. The time cost of distance correlation could be substantial when the sample size is just a few thousand. When the random variables are univariate, there exist efficient numerical algorithms of time complexity

$O(n \log n)$ [8]. However, for the multivariate random variables, we have not found any efficient algorithms in existing papers after an extensive literature survey.

Independence tests of multivariate random variables could have a wide range of applications. In many problem settings, as mentioned in [9], each experimental unit will be measured multiple times, resulting in multivariate data. Researchers are often interested in exploring potential relationships among subsets of these measurements. For example, some measurements may represent attributes of physical characteristics while others represent attributes of psychological characteristics. It may be of interest to determine whether there exists a relationship between the physical and psychological characteristics. A test of independence between pairs of vectors, where the vectors may have different dimensions and scales, becomes crucial. Moreover, the number of experimental units, or equivalently, sample size, could be massive, which requires the test to be computationally efficient. This work will meet the demands for numerically efficient independence tests of multivariate random variables.

The newly proposed test of independence between two (potentially multivariate) random variables X and Y works as follows. Firstly, both X and Y are randomly projected to one-dimensional spaces. Then the fast computing method for distance covariances between a pair of univariate random variables is adopted to compute for a surrogate distance covariance. The above two steps are repeated numerous times. The final estimate of the distance covariance is the average of all aforementioned surrogate distance covariances.

For numerical efficiency, we will show (in Theorem 3.1) that the newly proposed algorithm enjoys the $O(Kn \log n)$ computational complexity and $O(\max\{n, K\})$ memory requirement, where K is the number of random projections and n is the sample size. On the statistical efficiency, we will show (in Theorem 4.19) that the asymptotic power of the test of independence by utilizing the newly proposed statistics is as efficient as its original multivariate counterpart, which achieves the state-of-the-art rates.

The rest of this paper is organized as follows. In **Section 2**, we review the definition of distance covariance, its fast algorithm in univariate cases, and related distance-based independence tests. **Section 3** gives the detailed algorithm for distance covariance of random vectors and corresponding independence tests. In **Section 4**, we present some theoretical properties on distance covariance and the asymptotic distribution of the proposed estimator. In **Section 5**, we conduct numerical examples to compare our method against others in the existing literature. Some discussions are presented in **Section 6**. We conclude in **Section 7**. All technical proofs, as well as the formal presentation of algorithms, are relegated to the appendix when appropriate.

Throughout this paper, we adopt the following notations. We denote $c_p = \frac{\pi^{(p+1)/2}}{\Gamma((p+1)/2)}$ and $c_q = \frac{\pi^{(q+1)/2}}{\Gamma((q+1)/2)}$ as two constants, where $\Gamma(\cdot)$ denotes the Gamma function. We will also need the following constants: $C_p = \frac{c_1 c_{p-1}}{c_p} = \frac{\sqrt{\pi}\Gamma((p+1)/2)}{\Gamma(p/2)}$ and $C_q = \frac{c_1 c_{q-1}}{c_q} = \frac{\sqrt{\pi}\Gamma((q+1)/2)}{\Gamma(q/2)}$. For any vector v, let v^t denote its transpose.

2 REVIEW OF DISTANCE COVARIANCE: DEFINITION, FAST ALGORITHM, AND RELATED INDEPENDENCE TESTS

In this section, we review some related existing works. In **Section 2.1**, we recall the concept of distance variances and correlations, as well as some of their properties. In **Section 2.2**, we discuss the estimators of distance covariances and correlations, as well as their computation. We present their applications in the test of independence in **Section 2.3**.

2.1 Definition of Distance Covariances

Measuring and testing the dependency between two random variables is a fundamental problem in statistics. The classical Pearson's correlation coefficient can be inaccurate and even misleading when nonlinear dependency exists [6]. propose the novel measure–distance correlation–which is exactly zero if and only if two random variables are independent. A limitation is that if the distance correlation is implemented based on its original definition, the corresponding computational complexity can be as high as $O(n^2)$, which is not desirable when n is large.

We review the definition of the distance correlation in [6]. Let us consider two random variables $X \in \mathbb{R}^p$, $Y \in \mathbb{R}^q$, $p \geq 1, q \geq 1$. Let the complex-valued functions $\phi_{X,Y}(\cdot)$, $\phi_X(\cdot)$, and $\phi_Y(\cdot)$ be the characteristic functions of the joint density of X and Y, the density of X, and the density of Y, respectively. For any function ϕ, we denote $|\phi|^2 = \phi\bar{\phi}$, where $\bar{\phi}$ is the conjugate of ϕ; in words, $|\phi|$ is the magnitude of ϕ at a particular point. For vectors, let us use $|\cdot|$ to denote the Euclidean norm. In [6], the definition of distance covariance between random variables X and Y is

$$\mathcal{V}^2(X,Y) = \int_{\mathbb{R}^{p+q}} \frac{|\phi_{X,Y}(t,s) - \phi_X(t)\phi_Y(s)|^2}{c_p c_q |t|^{p+1}|s|^{q+1}} dtds, \tag{2.1}$$

where two constants c_p and c_q have been defined at the end of **Section 1**. The distance correlation is defined as

$$\mathcal{R}^2(X,Y) = \frac{\mathcal{V}^2(X,Y)}{\sqrt{\mathcal{V}^2(X,X)}\sqrt{\mathcal{V}^2(Y,Y)}}.$$

The following property has been established in the aforementioned paper.

Theorem 2.1. Suppose $X \in \mathbb{R}^p, p \geq 1$ and $Y \in \mathbb{R}^q, q \geq 1$ are two random variables, the following statements are equivalent:

1) X is independent of Y;
2) $\phi_{X,Y}(t, s) = \phi_X(t)\phi_Y(s)$, for any $t \in \mathbb{R}^p$ and $s \in \mathbb{R}^q$;
3) $\mathcal{V}^2(X,Y) = 0$;
4) $\mathcal{R}^2(X,Y) = 0$.

Given sample $(X_1, Y_1), \ldots, (X_n, Y_n)$, we can estimate the distance covariance by replacing the population characteristic function with the sample characteristic function: for $i = \sqrt{-1}, t \in \mathbb{R}^p, s \in \mathbb{R}^q$, we define

$$\hat{\phi}_X(t) = \frac{1}{n}\sum_{j=1}^{n} e^{iX_j^t t},$$
$$\hat{\phi}_Y(s) = \frac{1}{n}\sum_{j=1}^{n} e^{iY_j^t s}, \text{ and}$$
$$\hat{\phi}_{X,Y}(t,s) = \frac{1}{n}\sum_{j=1}^{n} e^{iX_j^t t + iY_j^t s}.$$

Consequently one can have the following estimator for $\mathcal{V}^2(X,Y)$:

$$\mathcal{V}_n^2(X,Y) = \int_{\mathbb{R}^{p+q}} \frac{|\hat{\phi}_{X,Y}(t,s) - \hat{\phi}_X(t)\hat{\phi}_Y(s)|^2}{c_p c_q |t|^{p+1}|s|^{q+1}} dt \cdot ds. \tag{2.2}$$

Note that the above formula is convenient to define a quantity, however, is *not* convenient for computation, due to the integration on the right-hand side. In the literature, other estimates have been introduced and will be presented in the following.

2.2 Fast Algorithm in the Univariate Cases

The paper [10] gives an equivalent definition for the distance covariance between random variables X and Y:

$$\begin{aligned}\mathcal{V}^2(X,Y) &= \mathbb{E}[d(X,X')d(Y,Y')] \\ &= \mathbb{E}[|X-X'||Y-Y'|] - 2\mathbb{E}[|X-X'||Y-Y''|] \\ &\quad + \mathbb{E}[|X-X'|]\mathbb{E}[|Y-Y'|],\end{aligned} \tag{2.3}$$

where the double centered distance $d(\cdot,\cdot)$ is defined as

$$\begin{aligned}d(X,X') = |X-X'| - \mathbb{E}_X[|X-X'|] - \mathbb{E}_{X'}[|X-X'|] \\ + \mathbb{E}[|X-X'|],\end{aligned}$$

where $\mathbb{E}_X$, $\mathbb{E}_{X'}$ and $\mathbb{E}$ are expectations over X, X' and (X, X'), respectively.

Motivated by the above definition, one can give an unbiased estimator for $\mathcal{V}^2(X,Y)$. The following notations will be utilized: for $1 \le i, j \le n$,

$$\begin{aligned} a_{ij} &= |X_i - X_j|, \quad & b_{ij} &= |Y_i - Y_j|, \\ a_{i\cdot} &= \sum_{l=1}^{n} a_{il}, \quad & b_{i\cdot} &= \sum_{l=1}^{n} b_{il}, \\ a_{\cdot\cdot} &= \sum_{k,l=1}^{n} a_{kl}, \quad \text{and} & b_{\cdot\cdot} &= \sum_{k,l=1}^{n} b_{kl}. \end{aligned} \tag{2.4}$$

It has been proven [8, 28] that

$$\begin{aligned}\Omega_n(X,Y) = \frac{1}{n(n-3)}\sum_{i\ne j} a_{ij}b_{ij} - \frac{2}{n(n-2)(n-3)}\sum_{i=1}^{n} a_{i\cdot}b_{i\cdot} \\ + \frac{a_{\cdot\cdot}b_{\cdot\cdot}}{n(n-1)(n-2)(n-3)}\end{aligned} \tag{2.5}$$

is an unbiased estimator of $\mathcal{V}^2(X,Y)$. In addition, a fast algorithm has been proposed [8] for the aforementioned sample distance covariance in the univariate cases with complexity order $O(n\log\ n)$ and storage $O(n)$. We list the result below for reference purpose.

Theorem 2.2. (Theorem 3.2 & Corollary 4.1 in [8]). Suppose $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_n \in \mathbb{R}$. The unbiased estimator Ω_n defined in (2.5) can be computed by an $O(n\log\ n)$ algorithm.

In addition, as a byproduct, the following result is established in the same paper.

Corollary 2.3. The quantity

$$\frac{a_{\cdot\cdot}b_{\cdot\cdot}}{n(n-1)(n-2)(n-3)} = \frac{\sum_{k,l=1}^{n} a_{kl}\sum_{k,l=1}^{n} b_{kl}}{n(n-1)(n-2)(n-3)}$$

can be computed by an $O(n\log\ n)$ algorithm.

We will use the above result in our test of independence. However, as far as we know, in the multivariate cases, there does not exist any work on the fast algorithm of the order of complexity $O(n\log\ n)$. This paper will fill in this gap by introducing an order $O(nK\log\ n)$ complexity algorithm in multivariate cases.

2.3 Distance Based Independence Tests

Ref. [6] proposed an independence test using the distance covariance. We summarize it below as a theorem, which serves as a benchmark. Our test will be aligned with the following one, except that we introduced a new test statistic, which can be more efficiently computed, and it has comparable asymptotic properties with the test statistic that is used below.

Theorem 2.4. ([6], Theorem 6). For potentially multivariate random variables X and Y, a prescribed level α_s, and sample size n, one rejects the independence if and only if

$$\frac{n\mathcal{V}_n^2(X,Y)}{S_2} > (\Phi^{-1}(1-\alpha_s/2))^2,$$

where $\mathcal{V}_n^2(X,Y)$ has been defined in (2.2), $\Phi(\cdot)$ denote the cumulative distribution function of the standard normal distribution and

$$S_2 = \frac{1}{n^4}\sum_{i,j=1}^{n}|X_i - X_j|\sum_{i,j=1}^{n}|Y_i - Y_j|.$$

Moreover, let $\alpha(X, Y, n)$ denote the achieved significance level of the above test. If $\mathbb{E}[|X| + |Y|] < \infty$, then for all $0 < \alpha_s < 0.215$, one can show the following:

$$\lim_{n\to\infty} \alpha(X,Y,n) \le \alpha_s, \text{ and}$$
$$\sup_{X,Y}\left\{\lim_{n\to\infty} \alpha(X,Y,n)\colon \mathcal{V}(X,Y) = 0\right\} = \alpha_s.$$

Note that the quantity $\mathcal{V}_n^2(X,Y)$ that is used above as in [6] differs from the one that will be used in our proposed method. As mentioned, we use the above as an illustration for distance-based tests of independence, as well as the theoretical/asymptotic properties that such a test can achieve.

3 NUMERICALLY EFFICIENT METHOD FOR RANDOM VECTORS

This section is made of two components. We present a random-projection-based distance covariance estimator that will be proven to be unbiased with a computational complexity that is $O(Kn\log\ n)$ in **Section 3.1**. In **Section 3.2**, we describe how the test of independence can be done by utilizing the above estimator. For users' convenience, stand-alone algorithms are furnished in the **Supplementary Appendix**.

3.1 Random Projection Based Methods for Approximating Distance Covariance

We consider how to use a fast algorithm for univariate random variables to compute or approximate the sample distance covariance of random vectors. The main idea works as follows: first, projecting the multivariate observations on some random directions; then, using the fast algorithm to compute the distance covariance of the projections; at last, averaging distance covariances from different projecting directions.

More specifically, our estimator can be computed as follows. For potentially multivariate $X_1, \ldots, X_n \in \mathbb{R}^p$ and $Y_1, \ldots, Y_n \in \mathbb{R}^q$, let K be a predetermined number of iterations, we do:

1) For each k $(1 \le k \le K)$, randomly generate u_k and v_k from $\text{Uniform}(\mathcal{S}^{p-1})$ and $\text{Uniform}(\mathcal{S}^{q-1})$, respectively. Here $\mathcal{S}^{p-1}$ and $\mathcal{S}^{q-1}$ are the unit spheres in $\mathbb{R}^p$ and $\mathbb{R}^q$, respectively.
2) Let $u_k^t X$ and $v_k^t Y$ denote the projections of X and Y to the space that are orthogonal to vectors u_k and v_k, respectively. That is we have

$$u_k^t X = (u_k^t X_1, \ldots, u_k^t X_n), \text{ and } v_k^t Y = (v_k^t Y_1, \ldots, v_k^t Y_n).$$

Note that samples $u_k^t X$ and $v_k^t Y$ are now univariate.

3) Utilize the fast (i.e., order $O(n\log\ n)$) algorithm that was mentioned in Theorem 2.2 to compute for the unbiased estimator in **Eq. 2.5** with respect to $u_k^t X$ and $v_k^t Y$. Formally, we denote

$$\Omega_n^{(k)} = C_p C_q \Omega_n (u_k^t X, v_k^t Y),$$

where C_p and C_q have been defined at the end of **Section 1**.

(4) The above three steps are repeated for K times. The final estimator is

$$\bar{\Omega}_n = \frac{1}{K} \sum_{k=1}^{K} \Omega_n^{(k)}. \tag{3.1}$$

To emphasize the dependency of the above quantity with K, we sometimes use a notation $\bar{\Omega}_{n,K} \triangleq \bar{\Omega}_n$.

See **Algorithm 1** in the **Supplementary Appendix** for a stand-alone presentation of the above method. In the light of Theorem 2.2, we can handily declare the following.

Theorem 3.1. For potentially multivariate $X_1, \ldots, X_n \in \mathbb{R}^p$ and $Y_1, \ldots, Y_n \in \mathbb{R}^q$, the order of computational complexity of computing the aforementioned $\bar{\Omega}_n$ is $O(Kn\log\ n)$ with storage $O(\max\{n,\ K\})$, where K is the number of random projections.

The proof of the above theorem is omitted because it is straightforward from Theorem 2.2. The statistical properties of the proposed estimator $\bar{\Omega}_n$ will be studied in the subsequent section (specifically in **Section 4.4**).

3.2 Test of Independence

By a later result (cf. Theorem 4.19), we can apply $\bar{\Omega}_n$ in the independence tests. The corresponding asymptotic distribution of the test statistic $\bar{\Omega}_n$ can be approximated by a $\text{Gamma}(\alpha, \beta)$ distribution with α and β given in **Eq. 4.7**. We can compute the significance level of the test statistic by permutation and conduct the independence test accordingly. Recall that we have potentially multivariate $X_1, \ldots, X_n \in \mathbb{R}^p$ and $Y_1, \ldots, Y_n \in \mathbb{R}^q$. Recall that K denotes the number of Monte Carlo iterations in our previous algorithm. Let α_s denote the prescribed significance level of the independence test. Let L denote the number of random permutations that we will adopt. We would like to test the null hypothesis $\mathcal{H}_0$—X and Y are independent—against its alternative. Recall $\bar{\Omega}_n$ is our proposed estimator in **Eq. 3.1**. The following algorithm describes a test of independence, which applies permutation to generate a threshold.

1) For each ℓ, $1 \le \ell \le L$, generate a random permutation of Y: $Y^{*,\ell} = (Y_1^*, \ldots Y_n^*)$;
2) Using the algorithm in **Section 3.1**, one can compute the estimator $\bar{\Omega}_n$ as in **Eq. 3.1** for X and $Y^{*,\ell}$; denote the outcome to be $V_\ell = \bar{\Omega}_n(X, Y^{*,\ell})$. Note under random permutations, X and $Y^{*,\ell}$ are independent.
3) The above two steps are executed for all $\ell = 1, \ldots, L$. One rejects $\mathcal{H}_0$ if and only if we have

$$\frac{1 + \sum_{\ell=1}^{L} I(\bar{\Omega}_n > V_\ell)}{1 + L} > \alpha_s.$$

See **Algorithm 2** in the **Supplementary Appendix** for a stand-alone description.

It is notified that one can use the approximate asymptotic distribution information to estimate a threshold in the independence test. The following describes such an approach. Recall that random vectors $X_1, \ldots, X_n \in \mathbb{R}^p$ and $Y_1, \ldots, Y_n \in \mathbb{R}^q$, number of random projections K, and a prescribed significance level α_s have been mentioned earlier.

1) For each k $(1 \le k \le K)$, randomly generate u_k and v_k from $\text{uniform}(\mathcal{S}^{p-1})$ and $\text{uniform}(\mathcal{S}^{q-1})$, respectively.
2) Use the fast algorithm in Theorem 2.2 to compute the following quantities:

$$\begin{aligned} \Omega_n^{(k)} &= C_p C_q \Omega_n (u_k^t X, v_k^t Y), \\ S_{n,1}^{(k)} &= C_p^2 C_q^2 \Omega_n (u_k^t X, u_k^t X) \Omega_n (v_k^t Y, v_k^t Y), \\ S_{n,2}^{(k)} &= C_p \frac{a_{\cdot\cdot}^{u_k}}{n(n-1)}, \quad S_{n,3}^{(k)} = C_q \frac{b_{\cdot\cdot}^{v_k}}{n(n-1)}, \end{aligned}$$

where C_p and C_q have been defined at the end of **Section 1** and in the last equation, the $a_{\cdot\cdot}^{u_k}$ and $b_{\cdot\cdot}^{v_k}$ are defined as follows:

$$a_{ij}^{u_k} = |u_k^t(X_i - X_j)|, \quad b_{ij}^{v_k} = |v_k^t(Y_i - Y_j)|,$$
$$a_{\cdot\cdot}^{u_k} = \sum_{k,l=1}^{n} a_{kl}^{u_k}, \quad b_{\cdot\cdot}^{v_k} = \sum_{k,l=1}^{n} b_{kl}^{v_k}.$$

3) For the aforementioned k, randomly generate u_k' and v_k' from uniform$(\mathcal{S}^{p-1})$ and uniform$(\mathcal{S}^{q-1})$, respectively. Use the fast algorithm that is mentioned in Theorem 2.2 to compute the following.

$$\Omega_{n,X}^{(k)} = C_p^2\Omega_n(u_k^tX, u'^t_kX), \quad \Omega_{n,Y}^{(k)} = C_p^2\Omega_n(v_k^tY, v'^t_kY).$$

where C_p and C_q have been defined at the end of **Section 1**.

4) Repeat the previous steps for all $k = 1, \ldots, K$. Then we compute the following quantities:

$$\bar{\Omega}_n = \frac{1}{K}\sum_{k=1}^{K}\Omega_n^{(k)}, \quad \bar{S}_{n,1} = \frac{1}{K}\sum_{k=1}^{K}S_{n,1}^{(k)}, \quad \bar{S}_{n,2} = \frac{1}{K}\sum_{k=1}^{K}S_{n,2}^{(k)},$$
$$\bar{S}_{n,3} = \frac{1}{K}\sum_{k=1}^{K}S_{n,3}^{(k)}, \quad \bar{\Omega}_{n,X} = \frac{1}{K}\sum_{k=1}^{K}\Omega_{n,X}^{(k)}, \quad \bar{\Omega}_{n,Y} = \frac{1}{K}\sum_{k=1}^{K}\Omega_{n,Y}^{(k)},$$
$$\alpha = \frac{1}{2}\frac{\bar{S}_{n,2}^2\bar{S}_{n,3}^2}{\frac{K-1}{K}\bar{\Omega}_{n,X}\bar{\Omega}_{n,Y} + \frac{1}{K}\bar{S}_{n,1}}, \tag{3.2}$$

$$\beta = \frac{1}{2}\frac{\bar{S}_{n,2}\bar{S}_{n,3}}{\frac{K-1}{K}\bar{\Omega}_{n,X}\bar{\Omega}_{n,Y} + \frac{1}{K}\bar{S}_{n,1}}. \tag{3.3}$$

5) Reject $\mathcal{H}_0$ if $n\bar{\Omega}_n + \bar{S}_{n,2}\bar{S}_{n,3} > \text{Gamma}(\alpha, \beta; 1-\alpha_s)$; otherwise, accept it. Here Gamma$(\alpha, \beta; 1-\alpha_s)$ is the $1-\alpha_s$ quantile of the distribution Gamma(α, β).

The above procedure is motivated by the observation that the asymptotic distribution of the test statistic $n\bar{\Omega}_n$ can be approximated by a Gamma distribution, whose parameters can be estimated by **Eq. 3.2** and **Eq. 3.3**. A stand-alone description of the above procedure can be found in **Algorithm 3** in the **Supplementary Appendix**.

4 THEORETICAL PROPERTIES

In this section, we establish the theoretical foundation of the proposed method. In **Section 4.1**, we study some properties of the random projections and the subsequent average estimator. These properties will be needed in studying the properties of the proposed estimator. We study the properties of the proposed distance covariance estimator (Ω_n) in **Section 4.2**, taking advantage of the fact that Ω_n is a U-statistic. It turns out that the properties of eigenvalues of a particular operator play an important role. We present the relevant results in **Section 4.3**. The main properties of the proposed estimator $(\bar{\Omega}_n)$ are presented in **Section 4.4**.

4.1 Using Random Projections in Distance-Based Methods

In this section, we will study some properties of distance covariances of randomly projected random vectors. We begin with a necessary and sufficient condition of independence.

Lemma 4.1. Suppose *u and v* are points on the hyper-spheres: $u \in \mathcal{S}^{p-1} = \{u \in \mathbb{R}^p: |u| = 1\}$ and $v \in \mathcal{S}^{q-1}$. We have

$$random\,vectors\,X \in \mathbb{R}^p\,and\,Y \in \mathbb{R}^q\,are\,independent$$

if and only if

$$\mathcal{V}^2(u^tX, v^tY) = 0, for\,any\,u \in \mathcal{S}^{p-1}, v \in \mathcal{S}^{q-1}.$$

The proof is relatively straightforward. We relegate a formal proof to the appendix. This lemma indicates that the independence is somewhat preserved under projections. The main contribution of the above result is to motivate us to think of using random projection, to reduce the multivariate random vectors into univariate random variables. As mentioned earlier, there exist fast algorithms of distance-based methods for univariate random variables.

The following result allows us to regard the distance covariance of random vectors of any dimension as an integral of distance covariance of univariate random variables, which are the projections of the aforementioned random vectors. The formulas in the following lemma provide the foundation for our proposed method: the distance covariances in the multivariate cases can be written as integrations of distance covariances in the univariate cases. our proposed method essentially adopts the principle of Monte Carlo to approximate such integrals. We again relegate the proof to the **Supplementary Appendix**.

Lemma 4.2. Suppose *u and v* are points on unit hyper-spheres: $u \in \mathcal{S}^{p-1} = \{u \in \mathbb{R}^p: |u| = 1\}$ and $v \in \mathcal{S}^{q-1}$. Let μ and ν *d*enote the uniform probability measure on $\mathcal{S}^{p-1}$ and $\mathcal{S}^{q-1}$, respectively. Then, we have for random vectors $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$,

$$\mathcal{V}^2(X, Y) = C_pC_q\int_{\mathcal{S}^{p-1}\times\mathcal{S}^{q-1}}\mathcal{V}^2(u^tX, v^tY)d\mu(u)d\nu(v),$$

where C_p and C_q are two constants that are defined at the end of Section 1. Moreover, a similar result holds for the sample distance covariance:

$$\mathcal{V}_n^2(X, Y) = C_pC_q\int_{\mathcal{S}^{p-1}\times\mathcal{S}^{q-1}}\mathcal{V}_n^2(u^tX, v^tY)d\mu(u)d\nu(v).$$

Besides the integral equations in the above lemma, we can also establish the following result for the unbiased estimator. Such a result provides the direct foundation of our proposed method. Recall that Ω_n, which is in **Eq. 2.5**, is an unbiased estimator of the distance covariance $\mathcal{V}^2(X, Y)$. A proof is provided in the **Supplementary Appendix**.

Lemma 4.3. Suppose u and v are points on the hyper-spheres: $u \in \mathcal{S}^{p-1} = \{u \in \mathbb{R}^p: |u| = 1\}$ and $v \in \mathcal{S}^{q-1}$. Let μ and ν denote the measure corresponding to the uniform densities on the surfaces $\mathcal{S}^{p-1}$ and $\mathcal{S}^{q-1}$, respectively. Then, we have

$$\Omega_n(X,Y) = C_p C_q \int_{\mathcal{S}^{p-1}\times\mathcal{S}^{q-1}} \Omega_n(u^t X, v^t Y) d\mu(u) d\nu(v),$$

where C_p and C_q are constants that were mentioned at the end of **Section 1**.

From the above lemma, recalling the design of our proposed estimator $\bar{\Omega}_n$ as in **Eq. 3.1**, it is straightforward to see that the proposed estimator $\bar{\Omega}_n$ is an unbiased estimator of $\Omega_n(X, Y)$. For completeness, we state the following without a proof.

Corollary 4.4. The proposed estimator $\bar{\Omega}_n$ in **Eq. 3.1**) is an unbiased estimator of the estimator $\Omega_n(X, Y)$ that was defined in **Eq. 2.5**.

Note that the estimator $\bar{\Omega}_n$ in **Eq. 3.1** evidently depends on the number of random projections K. Recall that to emphasize such a dependency, we sometimes use a notation $\bar{\Omega}_{n,K} \triangleq \bar{\Omega}_n$. The following concentration inequality shows the speed that $\bar{\Omega}_{n,K}$ can converge to Ω_n as $K \to \infty$.

Lemma 4.5. Suppose $\mathbf{E}[|X|^2] < \infty$ and $\mathbf{E}[|Y|^2] < \infty$. For any $\epsilon > 0$, we have

$$\mathbf{P}\left(|\bar{\Omega}_{n,K} - \Omega_n| > \epsilon\right) \le 2\exp\left\{-\frac{CK\epsilon^2}{Tr[\Sigma_X]Tr[\Sigma_Y]}\right\},$$

where Σ_X and Σ_Y are the covariance matrices of X and Y, respectively, $Tr[\Sigma_X]$ and $Tr[\Sigma_Y]$ are their matrix traces, and $C = \frac{2}{25C_p^2C_q^2}$ is a constant.

The proof is a relatively standard application of Hoeffding's inequality [11], which has been relegated to the appendix. The above lemma essentially indicates that the quantity $|\bar{\Omega}_{n,K} - \Omega_n|$ converges to zero at a rate no worse than $O(1/\sqrt{K})$.

4.2 Asymptotic Properties of the Sample Distance Covariance Ω_n

The asymptotic behavior of the sample distance covariance Ω_n in **Eq. 2.5** of this paper, has been studied in many places, seeing [5,8,10,12]. We found that it is still worthwhile to present them here, as we will use them to establish the statistical properties of our proposed estimator. The asymptotic distributions of Ω_n will be studied under two situations: 1) a general case and 2) when X and Y are assumed to be independent. We will see that the asymptotic distributions are different in these two situations.

It has been showed in ([8], Theorem 3.2) that Ω_n is a U-statistic. In the following, we state the result without formal proof. We will need the following function, denoted by h_4, which takes four pairs of input variables:

$$\begin{aligned} & h_4((X_1,Y_1),(X_2,Y_2),(X_3,Y_3),(X_4,Y_4)) \\ &= \frac{1}{4}\sum_{1\le i,j\le 4, i\ne j} |X_i - X_j||Y_i - Y_j| \\ &\quad - \frac{1}{4}\sum_{i=1}^{4}\left(\sum_{1\le j\le 4, j\ne i}|X_i - X_j| \sum_{1\le j\le 4, j\ne i}|Y_i - Y_j|\right) \\ &\quad + \frac{1}{24}\sum_{1\le i,j\le 4, i\ne j}|X_i - X_j|\sum_{1\le i,j\le 4, i\ne j}|Y_i - Y_j|. \end{aligned} \tag{4.1}$$

Note that the definition of h_4 coincides with Ω_n when the number of observations $n = 4$.

Lemma 4.6. (U-statistics). Let Ψ_4 denote all distinct 4-subset of $\{1, \ldots, n\}$ and let us define $X_\psi = \{X_i | i \in \psi\}$ and $Y_\psi = \{Y_i | i \in \psi\}$, then Ω_n is a U-statistic and can be expressed as

$$\Omega_n = \binom{n}{4}^{-1}\sum_{\psi\in\Psi_4} h_4\left(X_\psi, Y_\psi\right).$$

From the literature of the U-statistics, we know that the following quantities play critical roles. We state them here:

$$h_1((X_1,Y_1)) = \mathbb{E}_{2,3,4}[h_4((X_1,Y_1),(X_2,Y_2),(X_3,Y_3),(X_4,Y_4))],$$
$$h_2((X_1,Y_1),(X_2,Y_2)) = \mathbb{E}_{3,4}[h_4((X_1,Y_1),(X_2,Y_2),(X_3,Y_3),(X_4,Y_4))],$$
$$h_3((X_1,Y_1),(X_2,Y_2),(X_3,Y_3)) = \mathbb{E}_4[h_4((X_1,Y_1),(X_2,Y_2),(X_3,Y_3),(X_4,Y_4))],$$

where $\mathbb{E}_{2,3,4}$ stands for taking expectation over (X_2, Y_2), (X_3, Y_3) and (X_4, Y_4); $\mathbb{E}_{3,4}$ stands for taking expectation over (X_3, Y_3) and (X_4, Y_4); and $\mathbb{E}_4$ stands for taking expectation over (X_4, Y_4); respectively.

One immediate application of the above notations is the following result, which quantifies the variance of Ω_n. Since the formula is a known result, seeing ([13], Chapter 5.2.1 Lemma A), we state it without proof.

Lemma 4.7. (Variance of the U-statistic). The variance of Ω_n could be written as

$$\begin{aligned} Var(\Omega_n) &= \binom{n}{4}^{-1}\sum_{l=1}^{4}\binom{4}{l}\binom{n-4}{4-l}Var(h_l) \\ &= \frac{16}{n}Var(h_1) + \frac{240}{n^2}Var(h_1) + \frac{72}{n^2}Var(h_2) + O\left(\frac{1}{n^3}\right), \end{aligned}$$

where $O(\cdot)$ is the standard big O notation in mathematics.

From the above lemma, we can see that $\mathrm{Var}(h_1)$ and $\mathrm{Var}(h_2)$ play indispensable roles in determining the variance of Ω_n. The following lemma shows that under some conditions, we can ensure that $\mathrm{Var}(h_1)$ and $\mathrm{Var}(h_2)$ are bounded. A proof has been relegated to the appendix.

Lemma 4.8. If we have $\mathbb{E}[|X|^2] < \infty$, $\mathbb{E}[|Y|^2] < \infty$ and $\mathbb{E}[|X|^2|Y|^2] < \infty$, then we have $\mathrm{Var}(h_4) < \infty$. Consequently, we also have $\mathrm{Var}(h_1) < \infty$ and $\mathrm{Var}(h_2) < \infty$.

Even though as indicated in Lemma 4.7, the quantities $h_1(X_1, Y_1)$ and $h_2((X_1, Y_1), (X_2, Y_2))$ play important roles in determining the variance of Ω_n, in a generic case, they do not have a simple formula. The following lemma gives the generic formulas for $h_1(X_1, Y_1)$ and $h_2((X_1, Y_1), (X_2, Y_2))$. Its calculation can be found in the **Supplementary Appendix**.

Lemma 4.9. (Generic h_1 and h_2). In the general case, assuming (X_1, Y_1), (X, Y), (X', Y'), *and* (X'', Y'') are independent and identically distributed, we have

$$\begin{aligned} h_1((X_1, Y_1)) = \; & \frac{1}{2}\mathbb{E}[|X_1 - X'||Y_1 - Y'|] - \frac{1}{2}\mathbb{E}[|X_1 - X'||Y_1 \\ & - Y''|] + \frac{1}{2}\mathbb{E}[|X_1 - X'||Y - Y''|] - \frac{1}{2}\mathbb{E}[|X_1 \\ & - X'||Y' - Y''|] + \frac{1}{2}\mathbb{E}[|X - X''||Y_1 - Y'|] \\ & - \frac{1}{2}\mathbb{E}[|X' - X''||Y_1 - Y'|] + \frac{1}{2}\mathbb{E}[|X - X'||Y \\ & - Y'|] - \frac{1}{2}\mathbb{E}[|X - X'||Y - Y''|]. \end{aligned}$$

We have a similar formula for $h_2((X_1, Y_1), (X_2, Y_2))$ in (B.7). Due to its length, we do not display it here.

If one assumes that X and Y are independent, we can have a simpler formula for h_1, h_2, as well as their corresponding variances. We list the results below, with detailed calculations relegated to the appendix. One can see that under independence, the corresponding formulas are much simpler.

Lemma 4.10. When X and Y are independent, we have the following. For (X, Y) and (X', Y') that are independent and identically distributed as (X_1, Y_1) and (X_2, Y_2), we have

$$h_1((X_1, Y_1)) = 0, \tag{4.2}$$

$$\begin{aligned} h_2((X_1, Y_1), (X_2, Y_2)) = \; & \frac{1}{6}(|X_1 - X_2| - \mathbb{E}[|X_1 - X|] \\ & - \mathbb{E}[|X_2 - X|] + \mathbb{E}[|X - X'|]) \\ (|Y_1 - Y_2| - \mathbb{E}[|Y_1 - Y|] & - \mathbb{E}[|Y_2 - Y|] + \mathbb{E}[|Y - Y'|]), \end{aligned} \tag{4.3}$$

$$Var(h_2) = \frac{1}{36}\mathcal{V}^2(X, X)\mathcal{V}^2(Y, Y), \tag{4.4}$$

where $\mathbb{E}$ stands for the expectation operators with respect to X, X and X', Y, or Y and Y', whenever appropriate, respectively.

If we have $0 < \mathrm{Var}(h_1) < \infty$, it is known that the asymptotic distribution of Ω_n is normal, as stated in the following. Note that based on Lemma 4.10, X and Y cannot be independent; otherwise one should have $h_1 = 0$ almost surely. The following theorem is based on a known result on the convergence of U-statistics, seeing ([13], Chapter 5.5.1 Theorem A). We state it without a proof.

Theorem 4.11. Suppose $0 < Var(h_1) < \infty$ and $Var(h_4) < \infty$, then we have

$$\Omega_n \xrightarrow{P} \mathcal{V}^2(X, Y)$$

moreover, we have

$$\sqrt{n}(\Omega_n - \mathcal{V}^2(X, Y)) \xrightarrow{D} N(0, 16Var(h_1)), as\, n \to \infty.$$

When X and Y are independent, the asymptotic distribution of $\sqrt{n}\Omega_n$ is no longer normal. In this case, from Lemma 4.10, we have

$$h_1((X_1, Y_1)) = 0 \text{ almost surely, and } Var[h_1((X_1, Y_1))] = 0.$$

The following theorem, which applies a result in ([13], Chapter 5.5.2), indicates that $n\Omega_n$ converges to a weighted sum of (possibly infinitely many) independent χ_1^2 random variables.

Theorem 4.12. If X and Y are independent, the asymptotic distribution of Ω_n is

$$n\Omega_n \xrightarrow{D} \sum_{i=1}^{\infty} \lambda_i (Z_i^2 - 1) = \sum_{i=1}^{\infty} \lambda_i Z_i^2 - \sum_{i=1}^{\infty} \lambda_i,$$

where $Z_i^2 \sim \chi_1^2$ i.i.d, λ_i*'s* are the eigenvalues of operator G that is defined as

$$Gg(x_1, y_1) = \mathbb{E}_{x_2, y_2}[6h_2((x_1, y_1), (x_2, y_2))g(x_2, y_2)],$$

where function $h_2((\cdot, \cdot), (\cdot, \cdot))$ was defined in (4.3).

Proof. *T*he asymptotic distribution of Ω_n is from the result in ([13], Chapter 5.5.2).

See **Section 4.3** for more details on methods for computing the value of λ_i*'s*. In particular, we will show that we have $\sum_{i=1}^{\infty}\lambda_i = \mathbb{E}[|X - X'|]\mathbb{E}[|Y - Y'|]$ (Corollary 4.15) and $\sum_{i=1}^{\infty}\lambda_i^2 = \mathcal{V}^2(X, X)\mathcal{V}^2(Y, Y)$ (which is essentially from **Eq. 4.4** and Lemma 4.7).

4.3 Properties of Eigenvalues λ_i*'s*

From Theorem 4.12, we see that the eigenvalues λ_i*'s* play important role in determining the asymptotic distribution of Ω_n. We study its properties here. Throughout this subsection, we assume that X and Y are independent. Let us recall that the asymptotic distribution of sample distance covariance Ω_n,

$$n\Omega_n \xrightarrow{D} \sum_{i=1}^{\infty} \lambda_i (Z_i^2 - 1) = \sum_{i=1}^{\infty} \lambda_i Z_i^2 - \sum_{i=1}^{\infty} \lambda_i,$$

where λ_i*'s* are the eigenvalues of the operator G that is defined as

$$Gg(x_1, y_1) = \mathbb{E}_{x_2, y_2}[6h_2((x_1, y_1), (x_2, y_2))g(x_2, y_2)],$$

where function $\mathrm{h}_2((\cdot, \cdot), (\cdot, \cdot))$ was defined in **Eq. 4.3**. By definition, eigenvalues λ_1, λ_2, … corresponding to distinct solutions of the following equation

$$Gg(x_1, y_1) = \lambda g(x_1, y_1). \tag{4.5}$$

We now study the properties of λ_i*'s*. Utilizing Lemma 12 and Eq. 4.4 in [12], we can verify the following result. We give details of verifications in the **Supplementary Appendix**.

Lemma 4.13. Both of the following two functions are positive definite kernels:

$$h_X(X_1,X_2) = -|X_1 - X_2| + \mathbb{E}[|X_1 - X|] + \mathbb{E}[|X_2 - X|] - \mathbb{E}[|X - X'|]$$

and

$$h_Y(Y_1,Y_2) = -|Y_1 - Y_2| + \mathbb{E}[|Y_1 - Y|] + \mathbb{E}[|Y_2 - Y|] - \mathbb{E}[|Y - Y'|].$$

The above result gives us a foundation to apply the equivalence result that has been articulated thoroughly in [12]. *E*quipped with the above lemma, we have the following result, which characterizes a property o*f* λ_i's. The detailed proof can be found in the **Supplementary Appendix**.

Lemma 4.14. Suppose *{λ_1, λ_2, . . .}* are the set of eigenvalues of kernel $6h_2((x_1, y_1), (x_2, y_2))$, $\{\lambda_1^X, \lambda_2^X, \ldots\}$ and $\{\lambda_1^Y, \lambda_2^Y, \ldots\}$ are the sets of eigenvalues of the positive definite kernels h_X and h_Y, respectively. We have the following:

$$\{\lambda_1, \lambda_2, \ldots\} = \{\lambda_1^X, \lambda_2^X, \ldots\} \otimes \{\lambda_1^Y, \lambda_2^Y, \ldots\};$$

that is, each λ_i satisfying (4.5) can be written as, for some *j*, *j*′,

$$\lambda_i = \lambda_j^X \cdot \lambda_{j'}^Y$$

where $\lambda_j^X$and $\lambda_{j'}^Y$ are the eigenvalues corresponding to kernel functions $h_X(X_1, X_2)$ and $h_Y(Y_1, Y_2)$, respectively.

Above lemma implies that eigenvalues of h_2 could be obtained immediately after knowing the eigenvalues of h_X *and* h_Y. But, in practice, there usually does not exist analytic solution for even the eigenvalues of h_X *or* h_Y. Instead, given the observations $(X_1, \ldots, X_n)$ and $(Y_1, \ldots, Y_n)$, we can compute the eigenvalues of matrices $\tilde{K}_X = (h_X(X_i, X_j))_{n\times n}$ and $\tilde{K}_Y = (h_Y(Y_i, Y_j))_{n\times n}$ and use those empirical eigenvalues to approximate $\lambda_1^X, \lambda_2^X, \ldots$ and $\lambda_1^Y, \lambda_2^Y, \ldots$, and then consequently λ_1, λ_2, . . .

We end this subsection with the following corollary on the summations of eigenvalues, which is necessary for the proof of Theorem 4.12. The proof can be found in the **Supplementary Appendix**.

Corollary 4.15. The aforementioned eigenvalues $\lambda_1^X, \lambda_2^X, \ldots$ and $\lambda_1^Y, \lambda_2^Y, \ldots$ satisfy

$$\sum_{i=1}^{\infty} \lambda_i^X = \mathbb{E}[|X - X'|], and \sum_{i=1}^{\infty} \lambda_i^Y = \mathbb{E}[|Y - Y'|].$$

As a result, we have

$$\sum_{i=1}^{\infty} \lambda_i = \mathbb{E}[|X - X'|]\mathbb{E}[|Y - Y'|],$$

and

$$\sum_{i=1}^{\infty} \lambda_i^2 = \mathcal{V}^2(X,X)\mathcal{V}^2(Y,Y).$$

4.4 Asymptotic Properties of Averaged Projected Sample Distance Covariance $\bar{\Omega}_n$

We have reviewed the properties of the statistics Ω_n in a previous section (**Section 4.2**). The disadvantage of directly applying Ω_n (which is defined in **Eq. 2.5**) is that for multivariate *X* and *Y*, the implementation may require at least $O(n^2)$ operations. Recall that for univariate *X* and *Y*, an $O(n \log n)$ algorithm exists, cf. Theorem 2.2. The proposed estimator ($\bar{\Omega}_n$ in **Eq. 3.1**) is the averaged distance covariances, after randomly projecting *X* and *Y* to one-dimensional spaces, respectively. In this section, we will study the asymptotic behavior of $\bar{\Omega}_n$. It turns out that the analysis will be similar to the works in **Section 4.2**. The asymptotic distribution of $\bar{\Omega}_n$ will differ in two cases: (1) the general case and (2) the case when *X* and *Y* are independent.

As preparation for presenting the main result, we recall and introduce some notations. Recall the definition of $\bar{\Omega}_n$:

$$\bar{\Omega}_n = \frac{1}{K}\sum_{k=1}^{K} \Omega_n^{(k)},$$

where

$$\Omega_n^{(k)} = C_p C_q \Omega_n(u_k^t X, v_k^t Y)$$

and constants C_p, C_q have been defined at the end of **Section 1**. By Corollary 4.4, we have $\mathbb{E}[\Omega_n^{(k)}] = \Omega_n$, where $\mathbb{E}$ stands for the expectation with respect to the random projection. Note that from the work in**Section 4.2**, estimator $\Omega_n^{(k)}$ is a U-statistic. The following equation reveals that estimator $\bar{\Omega}_n$ is also a U-statistic,

$$\bar{\Omega}_n = \binom{n}{4}^{-1} \sum_{\psi\in\Psi_4} \frac{C_p C_q}{K} \sum_{k=1}^{K} h_4(u_k^t X_\psi, v_k^t Y_\psi) \triangleq \binom{n}{4}^{-1} \sum_{\psi\in\Psi_4} \bar{h}_4(X_\psi, Y_\psi),$$

where

$$\bar{h}_4(X_\psi, Y_\psi) = \frac{1}{K}\sum_{k=1}^{K} C_p C_q h_4(u_k^t X_\psi, v_k^t Y_\psi).$$

We have seen that quantities h_1 and h_2 play significant roles in the asymptotic behavior of statistic Ω_n. Let us define the counterpart notations as follows:

$$\begin{aligned}
&\bar{h}_1((X_1,Y_1)) \\
&= \mathbb{E}_{2,3,4}[\bar{h}_4((X_1,Y_1),(X_2,Y_2),(X_3,Y_3),(X_4,Y_4))] \triangleq \frac{1}{K}\sum\nolimits_{k=1}^{K} h_1^{(k)}, \\
&\bar{h}_2((X_1,Y_1),(X_2,Y_2)) \\
&= \mathbb{E}_{3,4}[\bar{h}_4((X_1,Y_1),(X_2,Y_2),(X_3,Y_3),(X_4,Y_4))] \triangleq \frac{1}{K}\sum\nolimits_{k=1}^{K} h_2^{(k)},
\end{aligned} \tag{4.6}$$

where $\mathbb{E}_{2,3,4}$ stands for taking expectation over (X_2, Y_2), (X_3, Y_3) and (X_4, Y_4); $\mathbb{E}_{3,4}$ stands for taking expectation over (X_3, Y_3) and (X_4, Y_4); as well as the following:

$$\begin{aligned}
h_1^{(k)} &= \mathbb{E}_{2,3,4}[C_p C_q h_4(u_k^t X_\psi, v_k^t Y_\psi)], \\
h_2^{(k)} &= \mathbb{E}_{3,4}[C_p C_q h_4(u_k^t X_\psi, v_k^t Y_\psi)].
\end{aligned}$$

In the general case, we do not assume that X and Y are independent. Let $U = (u_1, \ldots, u_K)$ and $V = (v_1, \ldots, v_K)$ denote the collection of random projections. We can write the variance of $\bar{\Omega}_n$as follows. The proof is an application of Lemma 4.7 and the law of total covariance. We relegate it to the **Supplementary Appendix**.

Lemma 4.16. Suppose $\mathbb{E}_{U,V}[\mathrm{Var}_{X,Y}(\bar{h}_1|U,V)] > 0$ and $\mathrm{Var}_{u,v}(\mathcal{V}^2(u^tX, v^tY)) > 0$, then, the variance of $\bar{\Omega}_n$ is

$$Var(\bar{\Omega}_n) = \frac{1}{K}Var_{u,v}(\mathcal{V}^2(u^tX, v^tY)) + \frac{16}{n}\mathbb{E}_{U,V}[Var_{X,Y}(\bar{h}_1|U,V)] + \frac{72}{n^2}\mathbb{E}_{U,V}[Var_{X,Y}(\bar{h}_2|U,V)] + O\left(\frac{1}{n^3}\right).$$

Equipped with the above lemma, we can summarize the asymptotic properties in the following theorem. We state it without proof as it is an immediate result from Lemma 4.16 as well as the contents in ([13], Chapter 5.5.1 Theorem A).

Theorem 4.17. Suppose $0 < \mathbb{E}_{U,V}[Var_{X,Y}(\bar{h}_1|U,V)] < \infty$, $\mathbb{E}_{U,V}[Var_{X,Y}(\bar{h}_4|U,V)] < \infty$. Also, let us assume that $K \to \infty$, $n \to \infty$, then we have

$$\bar{\Omega}_n \xrightarrow{P} \mathcal{V}^2(X,Y).$$

And, the asymptotic distribution of $\bar{\Omega}_n$ could differ under different conditions.

1) If $K \to \infty$ and $K/n \to 0$, then

$$\sqrt{K}(\bar{\Omega}_n - \mathcal{V}^2(X,Y)) \xrightarrow{D} N(0, Var_{u,v}(\mathcal{V}^2(u^tX, v^tY))).$$

2) If $n \to \infty$ and $K/n \to \infty$, then

$$\sqrt{n}(\bar{\Omega}_n - \mathcal{V}^2(X,Y)) \xrightarrow{D} N(0, 16\mathbb{E}_{U,V}[Var_{X,Y}(\bar{h}_1|U,V)]).$$

3) If $n \to \infty$ and $K/n \to C$, where C is some constant, then

$$\sqrt{n}(\bar{\Omega}_n - \mathcal{V}^2(X,Y)) \xrightarrow{D} N\left(0, \frac{1}{C}Var_{u,v}(\mathcal{V}^2(u^tX, v^tY)) + 16\mathbb{E}_{U,V}[Var_{X,Y}(\bar{h}_1|U,V)]\right).$$

Since our main idea is to utilize $\bar{\Omega}_n$ to approximate the quantity Ω_n, it is of interest to compare the asymptotic variance of Ω_n in Theorem 4.11 with the asymptotic variances in the above theorem. We present some discussions in the following remark.

Remark 4.18. Let us recall the asymptotic properties of Ω_n,

$$\sqrt{n}(\Omega_n - \mathcal{V}^2(X,Y)) \xrightarrow{D} N(0, 16Var(h_1)).$$

Then, we make the comparison in the following different scenarios.

1) If $K \to \infty$ and $K/n \to 0$, then the convergence rate of $\bar{\Omega}_n$ is much slower than Ω_n as $K \ll n$.
2) If $n \to \infty$ and $K/n \to \infty$, then the convergence rate of $\bar{\Omega}_n$ is the same with Ω_n and their variances is also the same
3) If $n \to \infty$ and $K/n \to C$, where C is some constant, then the convergence rate of $\bar{\Omega}_n$ is the same with Ω_n but the variance of $\bar{\Omega}_n$ is larger than that of Ω_n.

Generally, when X is not independent of Y, $\bar{\Omega}_n$ is as good as Ω_n in terms of convergence rate. However, in the independence test, the convergence rate of test statistics under the null hypotheses is of more interest. In the following context of this section, we will show that $\bar{\Omega}_n$ has the same convergence rate with Ω_n when X is independent of Y.

Now, let us consider the case that X and Y are independent. Similarly, by Lemma 4.10, we have

$$\bar{h}_1^{(k)} = 0, \bar{h}_1 = 0, \text{almost surely, and}, Var(\bar{h}_1) = 0.$$

And, by Lemma 4.1, we know that

$$\mathcal{V}^2(u^tX, v^tY) = 0, \forall u, v,$$

which implies

$$Var_{u,v}(\mathcal{V}^2(u^tX, v^tY)) = 0.$$

Therefore, we only need to consider $Var_{X,Y}(\bar{h}_2|U,V)$. Suppose (U, V) is given, a result in ([13], Chapter 5.5.2), together with Lemma 4.16, indicates that $n\bar{\Omega}_n$ converges to a weighted sum of (possibly infinitely many) independent χ_1^2 random variables. The proof can be found in the **Supplementary Appendix**.

Theorem 4.19If X and Y are independent, given the value of $U = (u_1, \ldots, u_K)$ and $V = (v_1, \ldots, v_K)$, the asymptotic distribution of $\bar{\Omega}_n$ is

$$n\bar{\Omega}_n \rightarrow {}_{D} \sum_{i=1}^{\infty} \bar{\lambda}_i(Z_i^2 - 1) = \sum_{i=1}^{\infty} \bar{\lambda}_i Z_i^2 - \sum_{i=1}^{\infty} \bar{\lambda}_i,$$

where $Z_i^2 \sim \chi_1^2$ i.i.d, and

$$\sum_{i=1}^{\infty} \bar{\lambda}_i = \frac{C_pC_q}{K} \sum_{k=1}^{K} \mathbb{E}[|u_k^t(X - X')|]\mathbb{E}[|v_k^t(Y - Y')|],$$

$$\sum_{i=1}^{\infty} \bar{\lambda}_i^2 = \frac{C_p^2C_q^2}{K^2} \sum_{k,k'=1}^{K} \mathcal{V}^2\left(u_k^tX, u_{k'}^tX\right)\mathcal{V}^2\left(v_k^tY, v_{k'}^tY\right).$$

Remark 4.20. Let us recall that if X and Y are independent, the asymptotic distribution of Ω_n is

$$n\Omega_n \xrightarrow{D} \sum_{i=1}^{\infty} \lambda_i(Z_i^2 - 1).$$

Theorem 4.19. shows that under the null hypotheses, $\bar{\Omega}_n$ enjoys the same convergence rate with Ω_n.

There usually does not exist a close-form expression for $\sum_{i=1}^{\infty}\bar{\lambda}_iZ_i^2$, but we can approximate it with the Gamma distribution whose first two moments matched. Thus, we have that $\sum_{i=1}^{\infty}\bar{\lambda}_iZ_i^2$ could be approximated *by Gamma(α, β)* with probability density function.

$$\frac{\beta^\alpha}{\Gamma(\alpha)}x^{\alpha-1}e^{-\beta x}, x > 0,$$

where

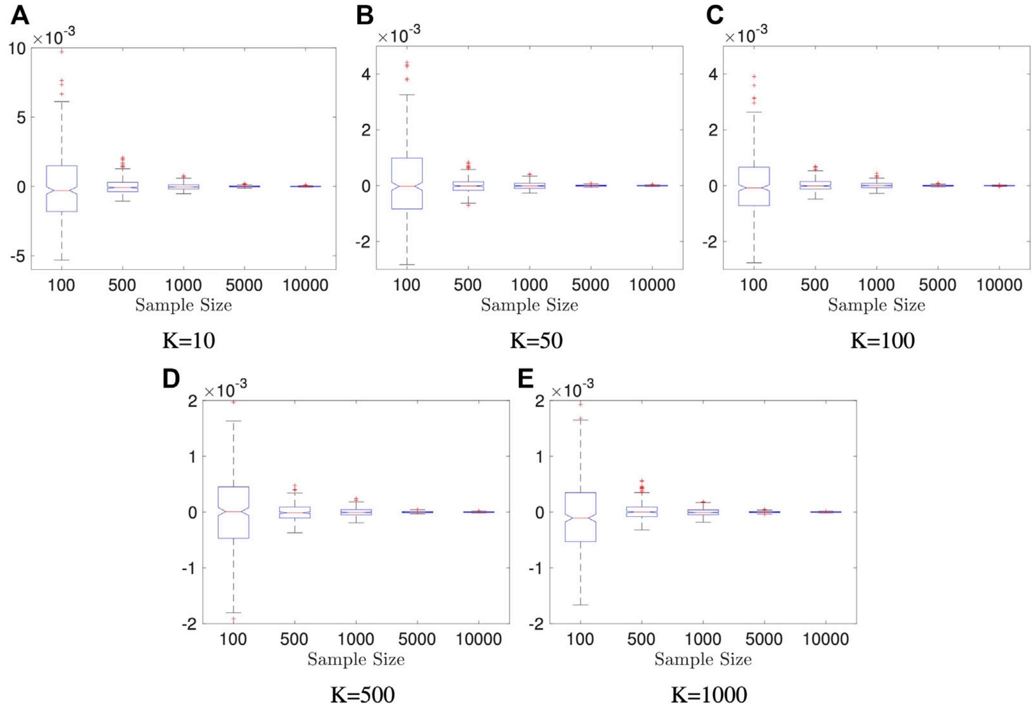


FIGURE 1 | Boxplots of estimators in Example 5.1: dimension of *X* and *Y* is fixed to be *p = q = 10*; the result is based on 400 repeated experiments. In each subplot, y-axis represents the value of distance covariance estimators.

$$\alpha = \frac{1}{2}\frac{\left(\sum_{i=1}^{\infty}\bar{\lambda}_i\right)^2}{\sum_{i=1}^{\infty}\bar{\lambda}_i^2}, \beta = \frac{1}{2}\frac{\sum_{i=1}^{\infty}\bar{\lambda}_i}{\sum_{i=1}^{\infty}\bar{\lambda}_i^2}. \tag{4.7}$$

See [14] **Section 3** for an empirical justification on this Gamma approximation. See [15] for a survey on different approximation methods of the weighted sum of the chi-square distribution.

The following result shows that both $\sum_{i=1}^{\infty}\bar{\lambda}_i$ and $\sum_{i=1}^{\infty}\bar{\lambda}_i^2$ could be estimated from data, see appendix for the corresponding justification.

Proposition 4.21. We can approximate $\sum_{i=1}^{\infty}\bar{\lambda}_i$ and $\sum_{i=1}^{\infty}\bar{\lambda}_i$ as follows:

$$\sum_{i=1}^{\infty}\bar{\lambda}_i \approx \frac{C_p C_q}{Kn^2(n-1)^2}\sum_{k=1}^{K} a_{\cdot\cdot}^{u_k} b_{\cdot\cdot}^{v_k},$$

$$\sum_{i=1}^{\infty}\bar{\lambda}_i^2 \approx \frac{K-1}{K}\Omega_n(X,X)\Omega_n(Y,Y) + \frac{C_p^2 C_q^2}{K}\sum_{k=1}^{K}\Omega_n(u_k^t X, u_k^t X)\Omega_n(v_k^t Y, v_k^t Y).$$

5 SIMULATIONS

Our numerical studies follow the works of [4,6,12]. In **Section 5.1**, we study how the performance of the proposed estimator is influenced by some parameters, including the sample size, the dimensionalities of the data, as well as the number of random projections in our algorithm. We also study and compare the computational efficiency of the direct method and the proposed method in **Section 5.2**. The comparison of the corresponding independence test with other existing methods will be included in **Section 5.3**.

5.1 Impact of Sample Size, Data Dimensions and the Number of Monte Carlo Iterations

In this part, we will use some synthetic data to study the impact of sample size n, data dimensions *(p, q)* and the number of the Monte Carlo iterations K on the convergence and test power of our proposed test statistic $\bar{\Omega}_n$. The significance level is set to be $\alpha_s = 0.05$. Each experiment is repeated for $N = 400$ times to get reliable mean and variance of estimators.

In first two examples, we fix data dimensions $p = q = 10$ and let the sample size n vary in 100, 500, 1000, 5000, 10000 and let the number of the Monte Carlo iterations K vary in 10, 50, 100, 500, and 1000. The data generation mechanism is described as follows, and it generates independent variables.

Example 5.1. We generate random vectors $X \in \mathbb{R}^{10}$ and $Y \in \mathbb{R}^{10}$. Each entry X_i follows *Unif(0, 1)*, independently. Each entry $Y_i = Z_i^2$, where Z_i follows *Unif(0, 1)*, independently.

See **Figure 1** for the boxplots of the outcomes of Example 5.1. In each subfigure, we fix the Monte Carlo Iteration Number K and let the number of observations n grow. It is worth noting that the scale of each subfigure could be different to display the entire boxplots. This experiment shows that the estimator converges to 0 regardless of the number of the Monte Carlo iterations. It also suggests that $K = 50$ Monte Carlo iterations should suffice in the independent cases.

The following example is to study dependent variables.

Example 5.2. We generate random vectors $X \in \mathbb{R}^{10}$ and $Y \in \mathbb{R}^{10}$ Each entry X_i follows *Unif(0, 1),* independently. Let Y_i denote the i-th entry of Y. We let $Y_1 = X_1^2$ and $Y_2 = X_2^2$ The rest entry of Y, $Y_i = Z_i^2$, $i = 3, \ldots, 10$, where Z_i follows *Unif(0, 1)*, independently.

See **Figure 2** for the boxplots of the outcomes of Example 5.2. In each subfigure, we fix the number of the Monte Carlo iterations K and let the number of observations n grow. This exam*ple* shows that if K is fixed, the variation of the estimator remains regardless of the sample size n. In the dependent cases, the number of the Monte Carlo iterations K plays a more important role in estimator convergence than sample size n.

The outcomes of Example 5.1 and 5.2 confirm the theoretical results that the proposed estimator converges to 0 as sample size n grows in the independent case, and converges to some nonzero number as the number of the Monte Carlo iterations K grows in the dependent case.

In the following two examples, we fix the sample size $n = 2000$ as we noticed that our method is more efficient than the direct method when n is large. We fix the number of the Monte Carlo iterations $K = 50$ and relax the restriction on the data dimensions to allow $p \neq q$ and let p, q vary in *(10, 50, 100, 500, 1000)*. We continue with an independent case as follows.

Example 5.3. We generate random vectors $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$. Each entry of X follows *Unif(0, 1),* independently. Each entry $Y_i = Z_i^2$, where Z_i follows *Unif(0, 1),* independently.

See **Figure 3** for the boxplots of the outcomes of Example 5.3. In each subfigure, we fix the dimension of X and let the dimension of Y grow. It is worth noting that the scale of each subfigure could be different to display the entire boxplots. It shows that the proposed estimator converges fairly fast in independent cases regardless of the dimension of the data.

The following presents a dependent case. In this case, only a small number of entries in X and Y are dependent, which means that the dependency structure between X and Y is low-dimensional though X or Y could be of high dimensions.

Example 5.4. We generate random vectors $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$. Each entry of X follow*s Unif(0, 1)*, independently. We let the first 5 entries of Y to be the square of the first 5 entries of X and let the rest entries of Y to be the square of some independent *Unif(0, 1)* random variables. Specifically, we let $Y_i = X_i^2, i = 1, \ldots, 5$, and, $Y_i = Z_i^2, i = 6, \ldots, q$, where Z_i*'s* are drawn independently from *Unif(0, 1).*

See **Figure 4** for the boxplots of the outcomes of Example 5.4. In each subfigure, we fix the dimension of X *a*nd let the dimension of Y grow. The test power of the proposed test against data dimensions can be seen in **Table 1**. It is worth noting that when the sample size is fixed, the test power of our method decays as the dimension of X and Y increase. We use the Direct Distance Covariance (DDC) defined in **Eq. 2.5** on the same data. As a contrast, the test power of DDC is 1.000 even $p = q = 1000$. This example raises a limitation of random projection: it may fail to detect the low dimensional dependency in high dimensional data. A possible remedy for this issue is performing dimension reduction before applying the proposed method. We do not research further along this direction since it is beyond the scope of this paper.

5.2 Comparison With Direct Method

In this section, we would like to illustrate the computational and space efficiency of the proposed method (RPDC). RPDC is much faster than the direct method (DDC, **Eq. 2.5**) when the sample size is large. It is worth noting that DDC is infeasible when the sample size is too large as its space complexity is $O(n^2)$. See **Table 2** for a comparison of computing time (unit: second) against the sample size n. This experiment is run on a laptop (MacBook Pro Retina, 13-inch, Early 2015, 2.7 GHz Intel Core i5, 8 GB 1867 MHz DDR3) with MATLAB R2016b (9.1.0.441655).

5.3 Comparison With Other Independence Tests

In this part, we compare the statistical test power of the proposed test (RPDC) with Hilbert-Schmidt Independence Criterion (HSIC) [4] as HSIC is gaining attention in machine learning and statistics communities. In our experiments, a Gaussian kernel with standard deviation $\sigma = 1$ is used for HSIC. We also compare with Randomized Dependence Coefficient (RDC) [16], which utilizes the technique of random projection as we do. Two classical tests for multivariate independence, which are described below, are included in the comparison as well as Direct Distance Covariance (DDC) defined in **Eq. 2.5**.

- Wilks Lambda (WL): the likelihood ratio test of hypotheses $\Sigma_{12} = 0$ with μ *un*known is based on

$$\frac{\det(S)}{\det(S_{11})\det(S_{22})} = \frac{\det(S_{22} - S_{21}S_{11}^{-1}S_{12})}{\det(S_{22})},$$

where $\det(\cdot)$ is the determinant, S, S_{11} and S_{22} denote the sample covariances of *(X, Y)*, X and Y, respectively, and S_{12} is the sample covariance $\widehat{\mathrm{Cov}}(X, Y)$. Under multivariate normality, the test statistic

$$W = -n \log\det(I - S_{22}^{-1}S_{21}S_{11}^{-1}S_{12})$$

has the Wilks Lambda distribution Λ(q, n − 1 − p, p), see [17].

- Puri-Sen (PS) statistics: [18], Chapter 8, proposed similar tests based on more general sample dispersion matrices T. In that test S, S_{11}, S_{12} *and* S_{22} are replaced by T, T_{11}, T_{12} *and* T_{22}, where T could be a matrix of Spearman's rank correlation statistics. Then, the test statistic becomes

$$W = -n \log\det(I - T_{22}^{-1}T_{21}T_{11}^{-1}T_{12})$$

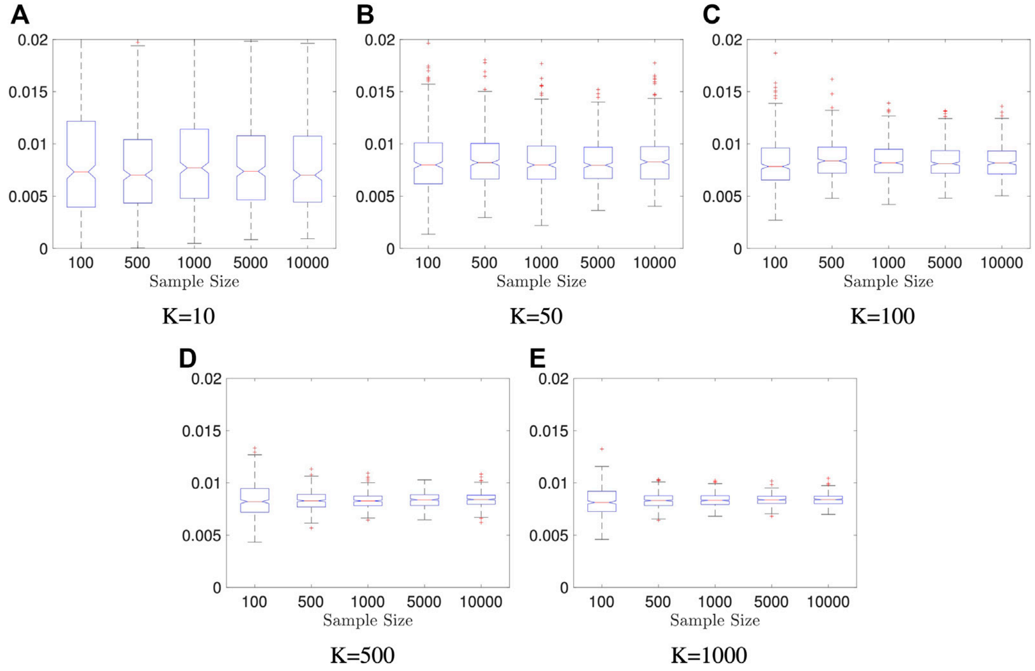


FIGURE 2 | Boxplots of our estimators in Example 5.2: dimension of *X* and *Y* is fixed to be *p* = *q* = *10*; the result is based on 400 repeated experiments. In each subplot, y-axis represents the value of distance covariance estimators.

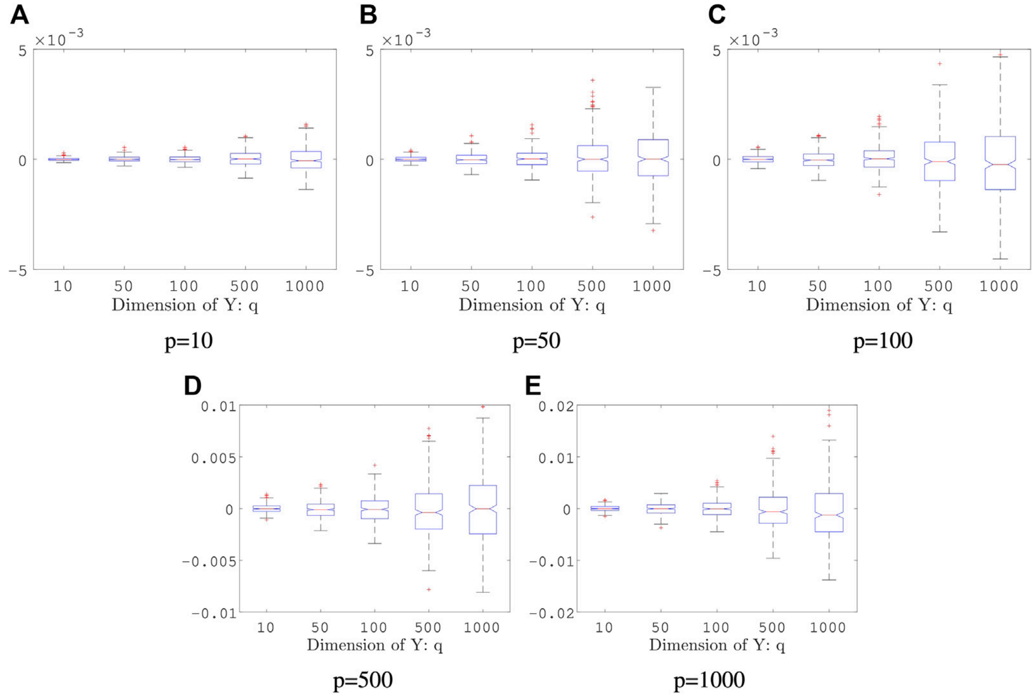


FIGURE 3 | Boxplot of Estimators in Example 5.3: both sample size and the number of Monte Carlo iterations is fixed, *n* = *2000, K* = *50*; the result is based on 400 repeated experiments. In each subplot, y-axis represents the value of distance covariance estimators.

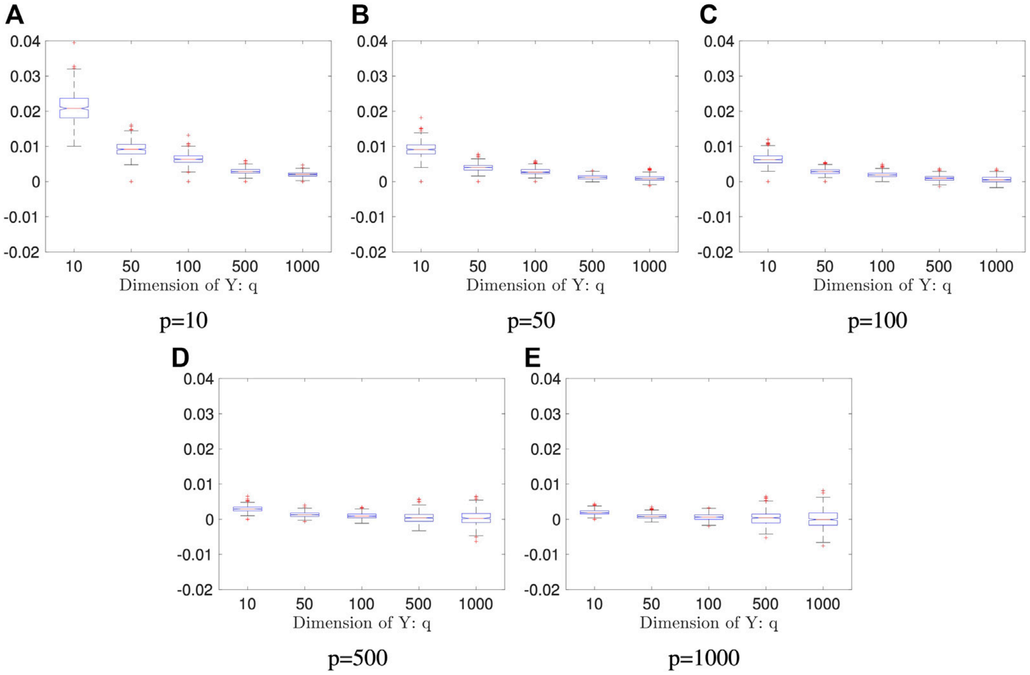


FIGURE 4 | Boxplots of the proposed estimators in Example 5.4: both sample size and the number of the Monte Carlo iterations are fixed: *n = 2000* and *K = 50*; the result is based on 400 repeated experiments. In each subplot, y-axis represents the value of distance covariance estimators.

TABLE 1 | Test Power in Example 5.4: this result is based 400 repeated experiments; the significance level is 0.05.

Dimension of *X*: p	**Dimension of *Y*: q**				
	10	**50**	**100**	**500**	**1000**
10	1.0000	1.0000	1.0000	1.0000	0.9975
50	1.0000	1.0000	1.0000	0.7775	0.4650
100	1.0000	1.0000	0.9925	0.4875	0.1800
500	0.9950	0.8150	0.4425	0.1225	0.0975
1000	0.9900	0.4000	0.2125	0.0900	0.0475

TABLE 2 | Speed Comparison: Direct Distance Covariance vs. Randomly Projected Distance Covariance.

Sample size	Ω_n	$\bar{\Omega}_n$
100	0.0043 (0.0047)	0.0207 (0.0037)
500	0.0210 (0.0066)	0.0770 (0.0086)
1000	0.0624 (0.0047)	0.1685 (0.0141)
2000	0.2349 (0.0133)	0.3568 (0.0169)
4000	0.9184 (0.0226)	0.7885 (0.0114)
8000	7.2067 (0.4669)	1.7797 (0.0311)
16000	—	3.7539 (0.0289)

This table is based on 100 repeated experiments, the dimension of X *and* Y *is fixed to be* p = q = 10 *and the number of Monte Carlo Iterations in RPDC is* K = 50. *The number outside of the brackets is the mean and the number inside of the brackets is the standard deviation.*

The critical values of the Wilks Lambda (WL) and Puri-Sen (PS) statistics are given by Bartlett's approximation ([19], Section 5.3.2b): if n is large and $p, q > 2$, then

$$-\left(n-\frac{1}{2}(p+q+3)\right)\log\det\left(I-S_{22}^{-1}S_{21}S_{11}^{-1}S_{12}\right)$$

has an approximation $\chi^2(pq)$ distribution.

The reference distributions of RDC and HSIC are approximated by 200 permutations. And the reference distributions of DDC and RPDC are approximated by Gamma Distribution. The significance level is set to be $\alpha_s = 0.05$ and each experiment is repeated for $N = 400$ times to get reliable type-I error/test power.

We start with an example that (X, Y) is multivariate normal. In this case, WL and PS are expected to be optimal as the assumptions of these two classical tests are satisfied. Surprisingly, DDC has comparable performance with the aforementioned two methods. RPDC can achieve satisfactory performance when the sample size is reasonably large.

Example 5.5. We set the dimension of the data to be $p = q = 10$. We generate random vectors $X \in \mathbb{R}^{10}$ and $Y \in \mathbb{R}^{10}$ from the standard multivariate normal distribution $\mathcal{N}(0, \mathbf{I}_{10})$. The joint distribution of (X, Y) is also normal and we have $Cor(X_i, Y_i) = \rho$, $i = 1, \ldots, 10$, and the rest correlation are all 0. We set the value of ρ to be 0 and 0.1 to represent independent and correlated scenarios, respectively. The sample size n is set to be from 100 to 1500 with an increment of 100.

Figure 5 plots the type-I error in subfigure (a) and test power in subfigure (b) against sample size. In the independence case $(\rho = 0.0)$, the type-I error of each test is always around the *significance* level α_s

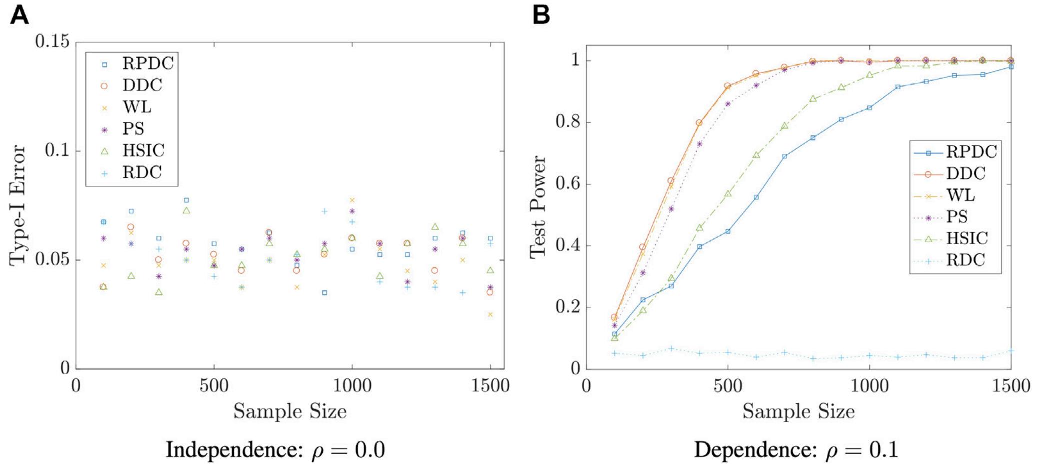


FIGURE 5 | Type-I Error/Test Power vs. Sample Size *n* in Example 5.5: the result is based on 400 repeated experiments.

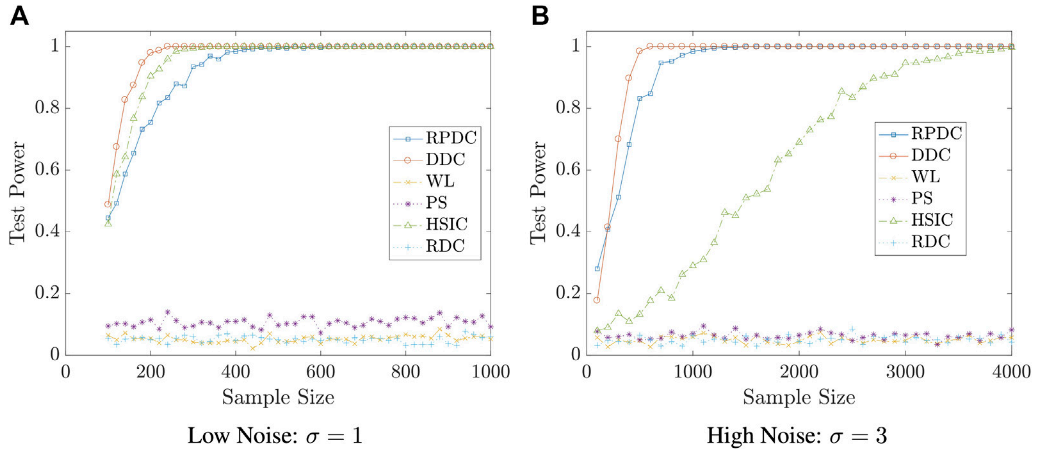


FIGURE 6 | Test Power vs. Sample Size *n* in Example 5.6: significance level is α_s = *0.05*; the result is based on *N* = *400* repeated experiments.

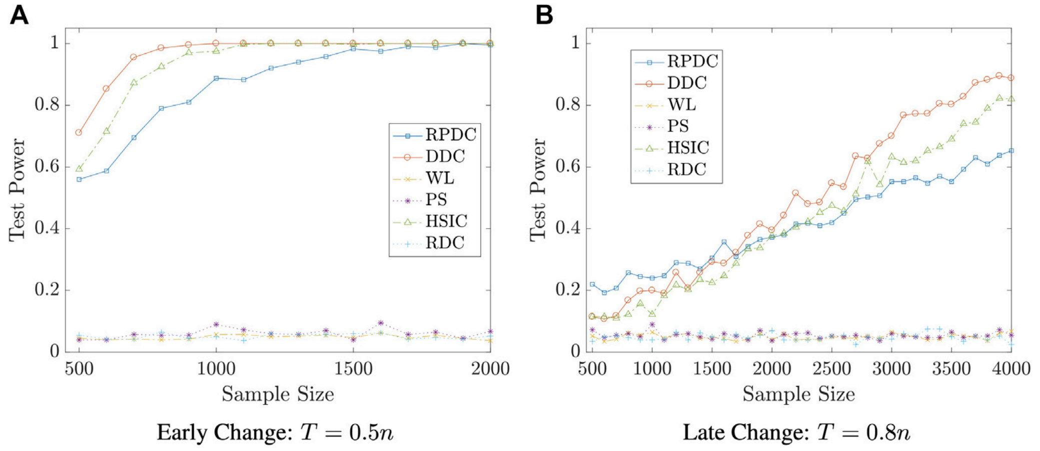
= *0.05*, which implies the Gamma approximation works well for asymptotic distributions. In the dependent case (ρ = 0.1), the overall performance of RPDC is close to HSIC and RPDC outperforms when the sample size is smaller and underperforms when the sample size is larger. Unfortunately, RDC's test power is insignificant.

Next, we compare those methods when *(X, Y)* is no longer multivariate normal and the dependency between *X* and *Y* is non-linear. We even add a noise term to compare their performance in both low and high noise-to-signal ratio scenarios. In this case, DDC and RPDC are much better than WL, PS, and RDC. The performance of HSIC is close to DDC and RPDC when the noise is low but much worse than those two when the noise is high.

Example 5.6. We set the dimension of data to be $p = q = 10$. We generate random vector $X \in \mathbb{R}^{10}$ from the standard multivariate normal distribution $\mathcal{N}(0, \mathbf{I}_{10})$. Let the i-th entry of Y be $Y_i = \log(X_i^2) + \epsilon_i, i = 1, \ldots, q$, where ϵ_i*'s* are independent random errors, $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. We set the value of σ to be 1 and 3 to represent low and high noise ratios, respectively. In the σ = *1* case, the sample size *n* is from 100 to 1000 with an increment 20; and in the σ = *3* case, the sample size *n is* from 100 to 4000 with an increment 100.

Figure 6 plots the test power of each test against sample size. In both low and high noise cases, none of WL, PS, and RDC has any test power. In the low noise case, all of RPDC, DDC, and HSIC have satisfactory test power *(> 0.9)* when the sample size is greater than 300. In the high noise case, RPDC and DDC could achieve more than 0.8 in test power once the sample size is greater than 500 while the test power of HSIC reaches 0.8 when the sample size is more than 2000.

In the following example, we generate the data similarly with Example 5.6 but the difference is that the dependency is changing over time. Specifically, *X and Y* are independent at the beginning but they become dependent after some time point. Since all those tests are invariant with the order of the observations, this experiment simply means that only a proportion of observations are dependent while the rest are not.



Early Change: $T = 0.5n$

Late Change: $T = 0.8n$

FIGURE 7 | Test Power vs. Sample Size n in Example 5.7: significance level is $\alpha_s = 0.05$; the result is based on $N = 400$ repeated experiments.

Example 5.7. We set the dimension of data to be p = q = 10. We generate random vector $X_t \in \mathbb{R}^{10}, t = 1, \ldots, n,$ from the standard multivariate normal distribution $\mathcal{N}(0, \mathbf{I}_{10})$. Let the i-th entry of Y_t be $Y_{t,i} = \log(Z_{t,i}^2) + \epsilon_{t,i}, t = 1, \ldots, T$ and $Y_{t,i} = \log(X_{t,i}^2) + \epsilon_{t,i}, t = T+1, \ldots, n,$ where Z_t *i.i.d.* $\sim \mathcal{N}(0, \mathbf{I}_{10})$ and $\epsilon_{t,i}$'s are independent random errors, $\epsilon_{t,i} \sim \mathcal{N}(0, 1)$. We set the value of T to be $0.5n$ and $0.8n$ to represent early and late dependency transition, respectively. In the early change case, the sample size n is from 500 to 2000 with an increment 100; and in the late change case, the sample size n is from 500 to 4000 with an increment 100.

Figure 7 plots the test power of each test against sample size. In both early and late change cases, none of WL, PS, and RDC has any test power. In the early change case, all of RPDC, DDC, and HSIC have satisfactory test power (> 0.9) when the sample size is greater than 1500. In the late change case, DDC and HSIC could achieve more than 0.8 in test power once sample size reaches 4000 while the test power of RPDC is only 0.6 when the sample size is 4000. As expected, the performance of DDC is better than RPDC in both cases and the performance of HSIC is between DDC and RPDC.

Remark 5.8. The examples in this subsection show that though RPDC underperforms DDC when the sample size is relatively small, RPDC could achieve the same test power with DDC when the sample size is sufficiently large. Thus, when the sample size is large enough, RPDC is superior to DDC because of its computational efficiency in both time and space.

6 DISCUSSIONS

6.1 A Discussion on the Computational Efficiency

We compare the computational efficiency of the proposed method (RPDC) and the direct method (DDC) in **Section 5.2.** We will discuss this issue here.

As $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$ are multivariate random variables, the effect of p and q on computing time could be significant when p and q are not negligible compared to sample size n. Now, we analyze the computational efficiency of DDC and RPDC by taking p and q into consideration. The computational complexity of DDC becomes $O(n^2(p + q))$ and that of RPDC becomes $O(nK(\log n + p + q))$. Let us denote the total number of operations in DDC by O_1 and that in RPDC by O_2. Then, there exist constants L_1 and L_2 such that

$$O_1 \approx L_1 n^2 (p+q), \text{and}\, O_2 \approx L_2 nK (\log n + p + q).$$

There is no doubt that O_2 will eventually be much less than O_1 as sample size n grows. Due to the complexity of the fast algorithm, we expect $L_2 > L_1$, which means the computing time of RPDC is even larger than DDC when the sample size is relatively small. Then, we need to study an interesting problem: what is the break-even point in terms of sample size n when RPDC and DDC have the same computing time?

Let $n_0 = n_0(p + q, K)$ denote the break-even point, which is a function of $p + q$ and number of Monte Carlo iterations K. For

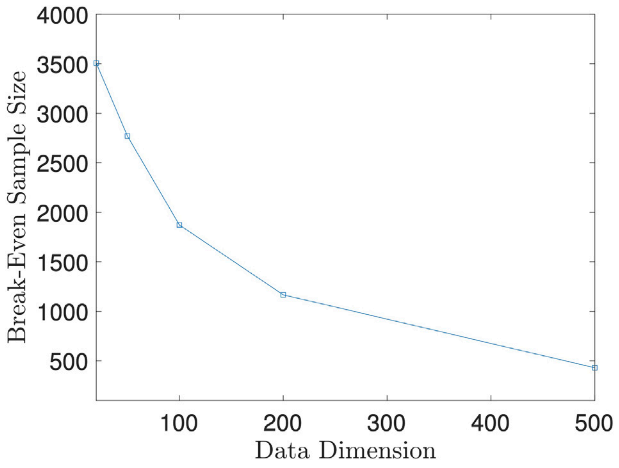


FIGURE 8 | Break-Even Sample Size n_0 against Data Dimension $p + q$. This figure is based on 100 repeated experiments.

simplicity, we fix $K = 50$ since 50 iterations could achieve satisfactory test power as we showed in Example 5.4. Then, n_0 becomes a function solely depending on $p + q$. Since it is hard to derive the close form of n_0, we derive it numerically instead. For fixed $p + q$, we let the sample size vary and record the difference between the running time of the two methods. Then, we fit the difference of running time against sample size with a smoothing spline. The root of this spline is the numerical value of n_0 at $p + q$.

We plot the n_0 against $p + q$ in **Figure 8**. As the figure shows, the break-even sample size decreases as the data dimension increases, which implies that our proposed method is more advantageous than the direct method when random variables are of high dimension. However, as shown in Example 5.4, the random projection-based method does not perform well when high dimensional data have a low dimensional dependency structure. We should be cautious to use the proposed method when the dimension is high.

6.2 Connections With Existing Literature

It turns out that distance-based methods are not the only choices in independence tests. See [20] and the references therein to see alternatives.

Our proposed method utilizes random projections, which bears a similarity with the randomized feature mapping strategy [21] that was developed in the machine learning community. Such an approach has been proven to be effective in kernel-related methods [22–26]. However, a closer examination will reveal the following difference: most of the aforementioned work is rooted in the Bochner's theorem [27] from harmonic analysis, which states that a continuous kernel in the Euclidean space is positive definite if and only if the kernel function is the Fourier transform of a non-negative measure. In this paper, we will deal with the distance function which is not a positive definite kernel. We will manage to derive a counterpart to the randomized feature mapping, which was the influential idea that has been used in [21].

Random projections have been used in [28] to develop a powerful two-sample test in high dimensions. They derived an asymptotic power function for their proposed test, and then provide sufficient conditions for their test to achieve greater power than other state-of-the-art tests. They then used the receiver operating characteristic (ROC) curves (that are generated from simulated data) to evaluate its performance against competing tests. The derivation of the asymptotic relative efficiency (ARE) is of its own interests. Despite the usage of random projection, the details of their methodology are very different from the one that is studied in the present paper.

Several distribution-free tests that are based on sample space partitions were suggested in [29] for univariate random variables. They proved that all suggested tests are consistent and showed the connection between their tests and the mutual information (MI). Most importantly, they derived fast (polynomial-time) algorithms, which are essential for large sample size, since the computational complexity of the naive algorithm is exponential in sample size. Efficient implementations of all statistics and tests described in the aforementioned paper are available in the R package HHG, which can be freely downloaded from the Comprehensive R Archive Network, http://cran.r-project.org/. Null tables can be downloaded from the first author's website.

Distance-based independence/dependence measurements sometimes have been utilized in performing a greedy feature selection, often *via* dependence maximization [8,30,31], and it has been effective on some real-world datasets. This paper simply mentions such a potential research line, without pursuing it.

Paper [32] derives an efficient approach to compute for the conditional distance correlations. We noted that there are strong resemblances between the distance covariances and its conditional version. The search for a potential extension of the work in this paper to conditional distance correlation can be a meaningful future topic of research.

Paper [33] provides some important insights into the power of distance covariance for multivariate data. In particular, they discover that distance-based independence tests have limiting power under some less common circumstances. As a remedy, they propose tests based on an aggregation of marginal sample distance and extend their approach to those based on Hilbert-Schmidt covariance and marginal distance/Hilbert-Schmidt covariance. It could be another interesting research direction but beyond the scope of this paper.

7 CONCLUSION

A significant contribution of this paper is we demonstrated that the multivariate variables in the independence tests need not imply the higher-order computational desideratum of the distance-based methods.

Distance-based methods are important in statistics, particularly in the test of independence. When the random variables are univariate, efficient numerical algorithms exist. It is an open question when the random variables are multivariate. This paper studies the random projection approach to tackle the above problem. It first turns the multivariate calculation problem into univariate calculation one via a random projection. Then they study how the average of those statistics out of the projected (therefore univariate) samples can approximate the distance-based statistics that were intended to use. Theoretical analysis was carried out, which shows that the loss of asymptotic efficiency (in the form of the asymptotic variance of the test statistics) is likely insignificant. The new method can be numerically much more efficient, when the sample size is large, which is well-expected under this information (or big-date) era. Simulation studies validate the theoretical statements. The theoretical analysis takes advantage of some newly available results, such as the equivalence of the distance-based methods with the reproducible kernel Hilbert spaces [12]. The numerical methods utilize a recently appeared algorithm in [8].

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

ACKNOWLEDGMENTS

This material was based upon work partially supported by the National Science Foundation under Grant DMS-1127914 to the Statistical and Applied Mathematical Sciences Institute. Any opinions, findings, and conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This work has also been partially supported by NSF grant DMS-1613152.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fams.2021.779841/full#supplementary-material

REFERENCES

1. David N, Yakir A, Hilary K, Sharon R, Peter J, Eric S, et al. Detecting Novel Associations in Large Data Sets. *Science* (2011) 334(6062):1518–24.
2. Schweizer B, Wolff EF. On Nonparametric Measures of Dependence for Random Variables. *Ann Stat* (1981) 879–85. doi:10.1214/aos/1176345528
3. Siburg KF, Stoimenov PA. A Measure of Mutual Complete Dependence. *Metrika* (2010) 71(2):239–51. doi:10.1007/s00184-008-0229-9
4. Gretton A, Bousquet O, Smola A, Schölkopf B. Measuring Statistical Dependence with hilbert-schmidt Norms. International Conference on Algorithmic Learning Theory. Springer (2005). p. 63–77. doi:10.1007/11564089_7
5. Székely GJ, Rizzo ML. Brownian Distance Covariance. *Ann Appl Stat* (2009) 3(4):1236–65. doi:10.1214/09-aoas312
6. GáborSzékely J, MariaRizzo L, Bakirov NK. Measuring and Testing Dependence by Correlation of Distances. *Ann Stat* (2007) 35(6):2769–94. doi:10.1214/009053607000000505
7. Matthew R, Nicolae DL. On Quantifying Dependence: a Framework for Developing Interpretable Measures. *Stat Sci* (2013) 28(1):116–30.
8. Huo X, Székely GJ. Fast Computing for Distance Covariance. *Technometrics* (2016) 58(4):435–47. doi:10.1080/00401706.2015.1054435
9. Taskinen S, Oja H, Randles RH. Multivariate Nonparametric Tests of independence. *J Am Stat Assoc* (2005) 100(471):916–25. doi:10.1198/016214505000000097
10. Lyons R. Distance Covariance in Metric Spaces. *Ann Probab* (2013) 41(5): 3284–305. doi:10.1214/12-aop803
11. Hoeffding W. Probability Inequalities for Sums of Bounded Random Variables. *J Am Stat Assoc* (1963) 58(301):13–30. doi:10.1080/01621459.1963.10500830
12. Sejdinovic D, Sriperumbudur B, Gretton A, Fukumizu K. Equivalence of Distance-Based and RKHS-Based Statistics in Hypothesis Testing. *Ann Stat* (2013) 41(5):2263–91. doi:10.1214/13-aos1140
13. Serfling RJ. *Approximation Theorems of Mathematical Statistics (Wiley Series in Probability and Statistics)*. Wiley-Interscience (1980).
14. George EP, Some Theorems on Quadratic Forms Applied in the Study of Analysis of Variance Problems, I. Effect of Inequality of Variance in the One-Way Classification. *Ann Math Stat* (1954) 25(2):290–302.
15. DeanBodenham A, Adams NM. A Comparison of Efficient Approximations for a Weighted Sum of Chi-Squared Random Variables. *Stat Comput pages* (2014) 1–12.
16. Lopez-Paz D, Hennig P, Schölkopf B. The Randomized Dependence Coefficient. In Advances in Neural Information Processing Systems, 1–9. 2013.
17. Wilks SS. On the Independence of K Sets of Normally Distributed Statistical Variables. *Econometrica* (1935) 3:309–26. doi:10.2307/1905324
18. Puri ML, Sen PK. *Nonparametric Methods in Multivariate Analysis. Wiley Series in Probability and Mathematical Statistics*. Probability and mathematical statistics. Wiley (1971).
19. Mardia KV, Bibby JM, Kent JT. *Multivariate Analysis*. New York, NY: Probability and Mathematical Statistics. Acad. Press (1982).
20. Lee K-Y, Li B, Zhao H. Variable Selection via Additive Conditional independence. *J R Stat Soc Ser B (Statistical Methodology)* (2016) 78(Part 5):1037–55. doi:10.1111/rssb.12150
21. Rahimi A, Recht B. Random Features for Large-Scale Kernel Machines. In Advances in Neural Information Processing Systems, 1177–84. 2007.
22. Achlioptas D, McSherry F, Schölkopf B. Sampling Techniques for Kernel Methods. In Annual Advances In Neural Information Processing Systems 14: Proceedings Of The 2001 Conference (2001).
23. Avrim B. Random Projection, Margins, Kernels, and Feature-Selection. *In Subspace, Latent Structure and Feature Selection*, 52–68. Springer, 2006.
24. Cai T, Fan J, Jiang T. Distributions of Angles in Random Packing on Spheres. *J Mach Learn Res* (2013) 14(1):1837–64.
25. Drineas P, Mahoney MW. On the Nyström Method for Approximating a Gram Matrix for Improved Kernel-Based Learning. *J Machine Learn Res* (2005) 6(Dec):2153–75.
26. Frieze A, Kannan R, Vempala S. Fast Monte-Carlo Algorithms for Finding Low-Rank Approximations. *J Acm* (2004) 51(6):1025–41. doi:10.1145/1039488.1039494
27. Rudin W. *Fourier Analysis on Groups*. John Wiley & Sons (1990).
28. Miles L, Laurent J, Wainwright J. A More Powerful Two-Sample Test in High Dimensions Using Random Projection. In Advances in Neural Information Processing Systems, 1206–14. 2011.
29. Heller R, Heller Y, Kaufman S, Brill B, Gorfine M. Consistent Distribution-free K-Sample and independence Tests for Univariate Random Variables. *J Machine Learn Res* (2016) 17(29):1–54.
30. Li R, Zhong W, Zhu L. Feature Screening via Distance Correlation Learning. *J Am Stat Assoc* (2012) 107(499):1129–39. doi:10.1080/01621459.2012.695654
31. Zhu L-P, Li L, Li R, Zhu L-X. Model-free Feature Screening for Ultrahigh-Dimensional Data. *J Am Stat Assoc* (2012).
32. Wang X, Pan W, Hu W, Tian Y, Zhang H. Conditional Distance Correlation. *J Am Stat Assoc* (2015) 110(512):1726–34. doi:10.1080/01621459.2014.993081
33. Zhu C, Zhang X, Yao S, Shao X Distance-based and Rkhs-Based Dependence Metrics in High Dimension. *Ann Stat* (2020) 48(6):3366–94. doi:10.1214/19-aos1934

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Applied Mathematics and Statistics

ORIGINAL RESEARCH
published: 20 January 2022
doi: 10.3389/fams.2021.784855

An Empirical Study of Graph-Based Approaches for Semi-supervised Time Series Classification

*Dominik Bünger, Miriam Gondos, Lucile Peroche and Martin Stoll**

Department of Mathematics, Chair of Scientific Computing, TU Chemnitz, Chemnitz, Germany

OPEN ACCESS

Edited by:
Stefan Kunis,
Osnabrück University, Germany

Reviewed by:
Tarek Emmrich,
Osnabrück University, Germany
Alex Jung,
Aalto University, Finland

Correspondence:
Martin Stoll
martin.stoll@
mathematik.tu-chemnitz.de

Specialty section:
This article was submitted to
Mathematics of Computation and
Data Science,
a section of the journal
Frontiers in Applied Mathematics and
Statistics

Received: *28 September 2021*
Accepted: *28 December 2021*
Published: *20 January 2022*

Citation:
Bünger D, Gondos M, Peroche L and
Stoll M (2022) An Empirical Study of
Graph-Based Approaches for
Semi-supervised Time Series
Classification.
Front. Appl. Math. Stat. 7:784855.
doi: 10.3389/fams.2021.784855

Time series data play an important role in many applications and their analysis reveals crucial information for understanding the underlying processes. Among the many time series learning tasks of great importance, we here focus on semi-supervised learning based on a graph representation of the data. Two main aspects are studied in this paper. Namely, suitable distance measures to evaluate the similarities between different time series, and the choice of learning method to make predictions based on a given number of pre-labeled data points. However, the relationship between the two aspects has never been studied systematically in the context of graph-based learning. We describe four different distance measures, including (Soft) DTW and MPDist, a distance measure based on the Matrix Profile, as well as four successful semi-supervised learning methods, including the recently introduced graph Allen–Cahn method and Graph Convolutional Neural Network method. We provide results for the novel combination of these distance measures with both the Allen-Cahn method and the GCN algorithm for binary semi-supervised learning tasks for various time-series data sets. In our findings we compare the chosen graph-based methods using all distance measures and observe that the results vary strongly with respect to the accuracy. We then observe that no clear best combination to employ in all cases is found. Our study provides a reproducible framework for future work in the direction of semi-supervised learning for time series with a focus on graph representations.

Keywords: semi-supervised learning, time series, graph Laplacian, Allen-Cahn equation, graph convolutional networks

1. INTRODUCTION

Many processes for which data are collected are time-dependent and as a result the study of time series data is a subject of great importance [1–3]. The case of time series is interesting for tasks such as anomaly detection [4], motif computation [5] or time series forecasting [6]. We refer to [7–10] for more general introductions.

We here focus on the task of classification of time series [11–16] in the context of semi-supervised learning [17, 18] where we want to label all data points[1] based on the fact that only a small portion of the data is already pre-labeled.

An example is given in **Figure 1** where we see some time series reflecting ECG (electrocardiogram) data and the classification into normal heartbeats on the one hand and myocardial infarction on the other hand. In our applications, we assume that only for some of the time series the corresponding class is known a priori. Our main contribution is to introduce a novel combination of incorporating the data into a *graph* and then incorporate this representation into several recently introduced methods for *semi-supervised learning*. For this, each time series becomes a node within a weighted undirected graph and the edge-weight is proportional to the similarity between different time series. Graph-based approaches have become a standard tool in many learning tasks (cf. [19–24] and the references mentioned therein). The matrix representation of the graph via its Laplacian [25] leads to studying the network using matrix properties. The Laplacian is *the* representation of the network that is utilized from machine learning to mathematical imaging. Recently, it has also been used network-Lasso-based learning approaches focusing on data with an inherent network structure, see e.g., [26, 27]. A very important ingredient in the construction of the Laplacian is the choice of the appropriate weight function. In many applications, the computation of the distance between time series or subsequences becomes a crucial task and this will be reflected in our choice of weight function. We consider several distance measures such as dynamic time warping DTW [28], soft DTW [29], and matrix profile [30].

We will embed these measures via the graph Laplacian into two different recently proposed semi-supervised learning frameworks. Namely, a diffuse interface approach that originates from material science [31] via the graph Allen-Cahn equation as well as a method based on graph convolutional networks [21]. Since these methods have originally been introduced outside of the field of time series learning, their relationship with time series distance measures has never been studied. Our goal is furthermore to compare these approaches with the well-known 1NN approach [11] and a simple optimization formulation solved relying on a linear system of equations. Our motivation follows that of [32, 33], where many methods for supervised learning in the context of time series were compared, namely that we aim to provide a wide-ranging overview of recent methods based on a graph representation of the data and combined with several distance measures.

We structure the paper as follows. In section 2, we introduce some basic notations and illustrate the basic notion of graph-based learning motivated with a clustering approach. In section 3, we discuss several distance measures with a focus on the well-known DTW measure as well as two recently emerged alternatives, i.e., Soft DTW and the MP distance. We use section 4 to introduce the two semi-supervised learning methods in more detail, followed by a shorter description of their well-known competitors. section 5 will allow us to compare the methods and study the hyperparameter selection.

2. BASICS

We consider discrete time series $\mathbf{x}_i$ given as a vector of real numbers of length m_i. In general, we allow for the time series to be of different dimensionality; later we often consider all $m_i = m$. We assume that we are given n time series $\mathbf{x}_i \in \mathbb{R}^{m_i}$. The goal of a classification task is to group the n time series into a number k of different *clusters* C_j with $j = 1, \ldots, k$. In this paper we focus on the task of semi-supervised learning [17] where only some of the data are already labeled but we want to classify all available data simultaneously. Nevertheless, we review some techniques for unsupervised learning first as they deliver useful terminology. As such the *k-means* algorithm is a prototype-based[2] clustering algorithm that divides the given data into a predefined number of k clusters [34]. The idea behind k-means is rather simple as the cluster centroids are repeatedly updated and the data points are assigned to the nearest centroid until the centroids and data points have converged. Often the termination condition is not handled that strictly. For example, the method can be terminated when only 1% of the points change clusters. The starting classes are often chosen at random but can also be assigned in a more systematic way by calculating the centers first and then assign the points to the nearest center. While k-means remains very popular it also has certain weaknesses coming from its minimization of the sum of squared errors loss function [35]. We discuss this method in some detail here to point out the main mechanism and this is based on assigning points to clusters and hence the cluster centroids based on the distance being the Euclidean norm, which would also be done when k-means is applied to time series. As a result the clusters might not capture the shape of the data manifold as illustrated in a simple two-dimensional example shown in **Figure 2**. In comparison, the alternative method shown, i.e., a spectral clustering technique, performs much better. We briefly discuss this method next as it forms the basis of the main techniques introduced in this paper.

2.1. Graph Laplacian and Spectral Clustering

As we illustrated in **Figure 2** the separation of the data into two-classes is rather difficult for k-means as the centroids are based on a 2-norm minimization. One alternative to k-means is based on interpreting the data points as nodes in a graph. For this, we assume that we are given data points $x_1, ..., x_n$ and some measure of similarity [23]. We define the weighted undirected similarity graph $G = (V, E)$ with the *vertex* or *node* set V and the edge set E. We view the data points $\mathbf{x}_i$ as vertices, $V = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, and if two nodes $(\mathbf{x}_i, \mathbf{x}_j)$ have a positive similarity function value, they are connected by an edge with weight w_{ij} equal to that similarity. With this reformulation of the data we turn the clustering problem into a graph partitioning problem where we want to cut the graph into two or possibly more classes.

[1] We here view one time-series as a data point and the feature vector for this data point is the vector with the associated data collected in a vector.

[2] Here the prototype of the cluster is the centroid.

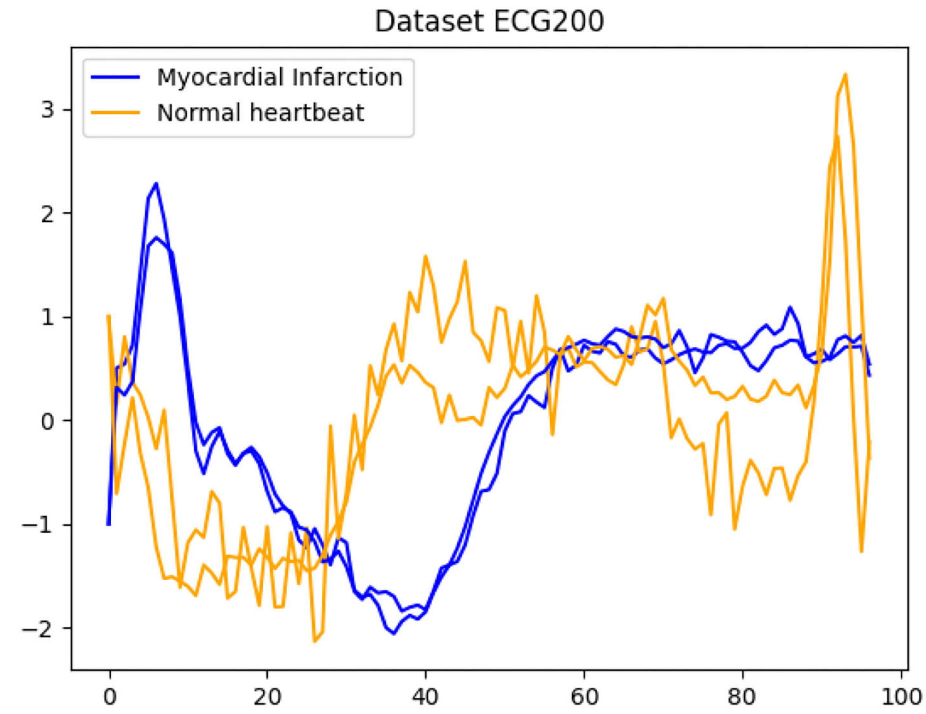


FIGURE 1 | A typical example for time series classification. Given the dataset ECG200, the goal is to automatically separate all time series into the classes *normal heartbeats* and *myocardial infarction*.

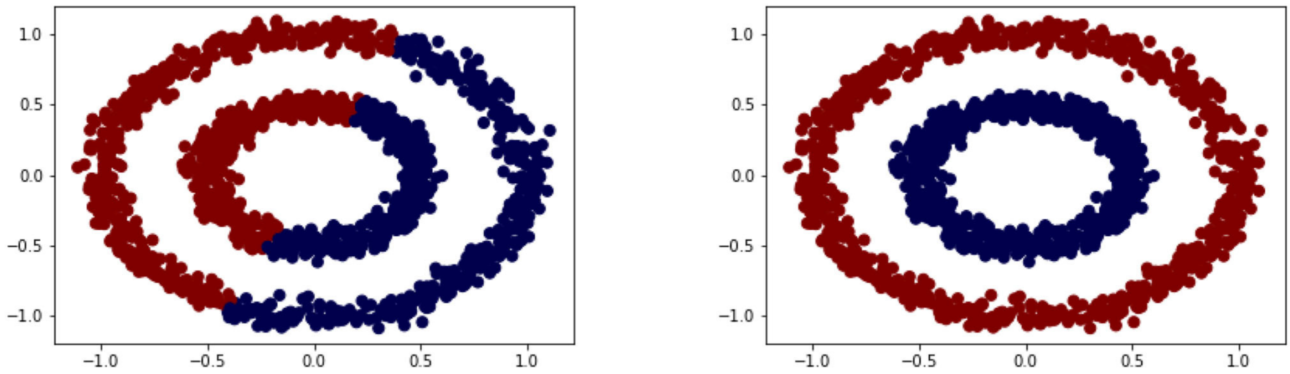


FIGURE 2 | Clustering based on original data via k-means **(left)** vs. transformed data via spectral clustering **(right)**.

This is usually done in such a way that the weight of the edges across the partition is minimal.

We collect all edge weights in the *adjacency matrix* $W = (w_{ij})_{i,j=1,\ldots,n}$. The degree of a vertex $\mathbf{x}_i$ is defined as $d_i = \sum_{j=1}^{n} w_{ij}$ and the degree matrix D is the diagonal matrix holding all n node degrees. In our case we use a fully connected graph with the *Gaussian similarity function*

$$w(\mathbf{x}_i, \mathbf{x}_j) = \exp\Big(-\frac{\text{dist}(\mathbf{x}_i, \mathbf{x}_j)^2}{\sigma^2}\Big), \tag{1}$$

where σ is a scaling parameter and $\text{dist}(\mathbf{x}_i, \mathbf{x}_j)$ is a particular distance function such as the Euclidean distance $\text{dist}(\mathbf{x}_i, \mathbf{x}_j) := \|\mathbf{x}_i - \mathbf{x}_j\|^2$. Note that for similar nodes, the value of the *distance* function is smaller than it would be for dissimilar nodes while the *similarity* function is relatively large.

We now use both the degree and weight matrix to define the *graph Laplacian* as $L = D - W$. Often the *symmetrically normalized Laplacian* defined via

$$L_{\text{sym}} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \tag{2}$$

provides better clustering information [23]. It has some very useful properties that we will exploit here. For example, given a non-zero vector $u \in \mathbb{R}^n$ we obtain the energy term

$$u^{\mathsf{T}} L_{\text{sym}} u = \frac{1}{2} \sum_{i,j} w_{ij} \left(\frac{u_i}{\sqrt{d_i}} - \frac{u_j}{\sqrt{d_j}} \right)^2 . \tag{3}$$

Using this it is easy to see that L_{sym} is positive semi-definite with non-negative eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$. The main advantage of the graph Laplacian is that based on its spectral information one can usually rely on transforming the data into a space where they are easier to separate [23, 25, 36]. As a result one typically requires the spectral information corresponding to the smallest eigenvalues of L_{sym}. The most famed eigenvector is the *Fiedler vector*, i.e., the eigenvector corresponding to the first non-zero eigenvalue, which is bound to have a sign change and as a result can be used for binary classification. The weight function (1) is also found in kernel methods [37, 38] when the radial basis kernel is applied.

2.2. Self-Tuning

In order to improve the performance of the methods based on the graph Laplacian, tuning the parameter σ is crucial. While hyperparameter tuning based on a grid search or cross validation is certainly possible we also consider a σ that adapts to the given data. For spectral clustering, such a procedure was introduced in [39]. Here we use this technique to learning with time series data. For each time series $\mathbf{x}_i$ we assume a local scaling parameter σ_i. As a result, we have the generalized square distance as

$$\frac{\text{dist}(\mathbf{x}_i, \mathbf{x}_j)}{\sigma_i} \frac{\text{dist}(\mathbf{x}_i, \mathbf{x}_j)}{\sigma_j} = \frac{\text{dist}(\mathbf{x}_i, \mathbf{x}_j)^2}{\sigma_i \sigma_j} \tag{4}$$

and this gives the entries of the adjacency matrix W via

$$w_{i,j} = \exp\left(-\frac{\text{dist}(\mathbf{x}_i, \mathbf{x}_j)^2}{\sigma_i \sigma_j} \right). \tag{5}$$

The authors in [39] choose σ_i as the distance to the K-th nearest neighbor of $\mathbf{x}_i$ where K is a fixed parameter, e.g., $K = 9$ is used in [31].

In section 5, we will explore several different values for K and their influence on the classification behavior.

3. DISTANCE MEASURES

We have seen from the definition of the weight matrix that the Laplacian depends on the choice of distance measure $\text{dist}(\mathbf{x}_i, \mathbf{x}_j)$. If all time series are of the same length then the easiest distance measure would be a Euclidean distance, which especially for large n is fast to compute. This makes the Euclidean distance incredibly popular but it suffers from being sensitive to small shifts in the time series. As a result we discuss several popular and efficient methods for different distance measures. Our focus is to illustrate in an empirical study how the choice of distance measure impacts the performance of graph-based learning and to provide further insights for future research (cf. [40]).

3.1. Dynamic Time Warping

We first discuss the distance measure of Dynamic Time Warping (DTW, [28]). By construction, DTW is an algorithm to find an optimal alignment between time series.

In the following, we adapt the notation of [28] to our case. Consider two time series $\mathbf{x}$ and $\tilde{\mathbf{x}}$ of lengths m and $\tilde{m}$, respectively, with entries $x_i, \tilde{x}_i \in \mathbb{R}$ for $i = 1, \ldots, m$ and $j = 1, \ldots, \tilde{m}$. We obtain the local cost matrix $C \in \mathbb{R}^{m \times \tilde{m}}$ by assembling the local differences for each pair of elements, i.e., $C_{ij} = |x_i - \tilde{x}_j|$.

The DTW distance is defined via $(m, \tilde{m})$-*warping paths*, which are sequences of index tuples $p = \big((i_1, j_1), ..., (i_L, j_L)\big)$ with boundary, monotonicity, and step size conditions

$$1 = i_1 \leq i_2 \leq \ldots \leq i_L = m, \quad 1 = j_1 \leq j_2 \leq \ldots \leq \tilde{m},$$
$$(i_{\ell+1} - i_\ell,\ j_{\ell+1} - j_\ell) \in \{(1,0), (0,1), (1,1)\} \quad (\ell = 1, \ldots, L-1).$$

The total cost of such a path with respect to $\mathbf{x}, \tilde{\mathbf{x}}$ is defined as

$$c_p(\mathbf{x}, \tilde{\mathbf{x}}) = \sum_{\ell=1}^{L} |x_{i_\ell} - \tilde{x}_{j_\ell}|.$$

The DTW distance is then defined as the minimum cost of any warping path:

$$\text{DTW}(\mathbf{x}, \mathbf{y}) := \min\{c_p(\mathbf{x}, \mathbf{y}) \mid p \text{ is a (m,}\tilde{m}\text{)-warping path}\}. \tag{6}$$

Both the warping and the warping path are illustrated in **Figure 3**.

Computing the optimal warping path directly quickly becomes infeasible. However, we can use dynamic programming to evaluate the accumulated cost matrix D recursively via

$$D(i,j) := |x_i - \tilde{x}_j| + \min\{D(i, j-1), D(i-1, j), D(i-1, j-1)\}. \tag{7}$$

The actual DTW distance is finally obtained as

$$\text{DTW}(\mathbf{x}, \mathbf{y}) = D(m, \tilde{m}). \tag{8}$$

The DTW method is a heavily used distance measure for capturing the sometimes subtle similarities between time series. In the literature it is typically stated that the computational cost of DTW being prohibitively large. As a result one is interested in accelerating the DTW algorithm itself. One possibility arises from imposing additional constraints (cf. [28, 41]) such as the Sakoe-Chiba Band and the Itakura parallelogram as these simplify the identification of the optimal warping path. While these are appealing concepts the authors in [42] observe that the well-known FastDTW algorithm [41] is in fact slower than DTW. For our purpose we will hence rely on DTW and in particular on the implementation of DTW provided via https://github.com/wannesm/dtaidistance. We observe that for this implementation of DTW indeed FastDTW is outperformed frequently.

3.2. Soft Dynamic Time Warping

Based on a slight reformulation of the above DTW scheme, we want to look at another time series distance measure, the *Soft Dynamic Time Warping* (Soft DTW). It is an extension of DTW designed allowing a differentiable loss function and it was

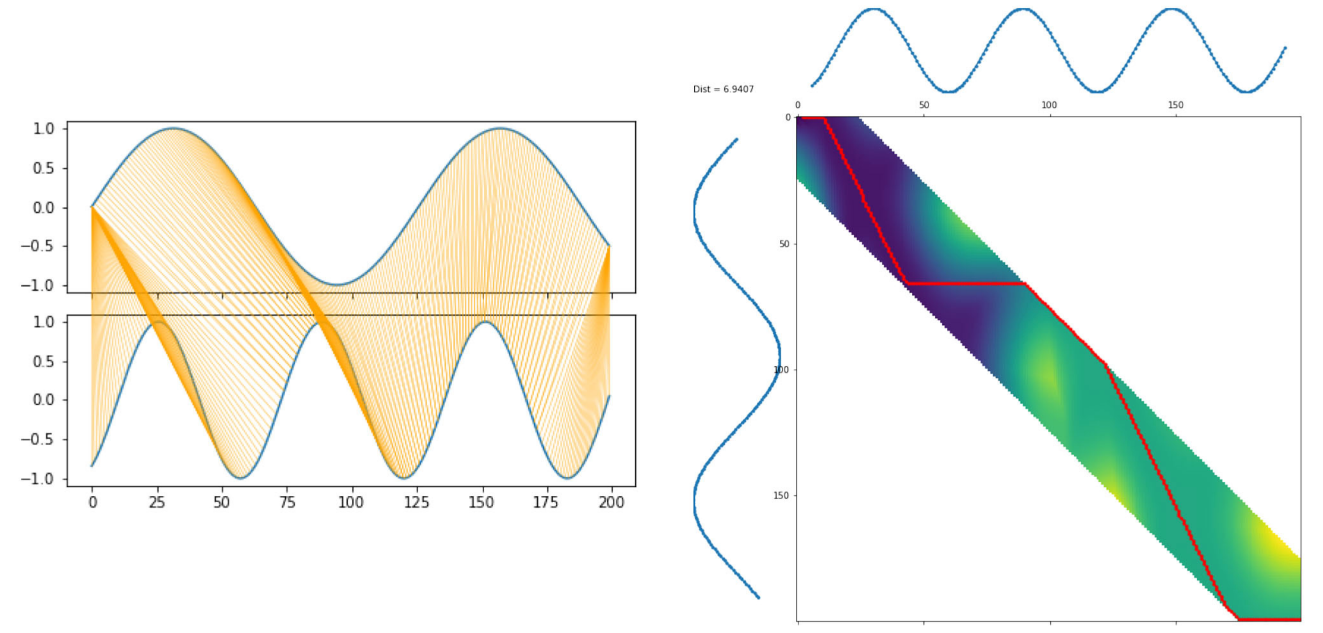


FIGURE 3 | DTW warping **(left)** and warpings paths **(right)**.

introduced in [29, 43]. We again start from the cost matrix C with $C(i,j) = |x_i - \tilde{x}_j|$ for time series $\mathbf{x}$ and $\tilde{\mathbf{x}}$. Each warping path can equivalently be described by a matrix $A \in \{0,1\}^{m\times\tilde{m}}$ with the following condition: The ones in A form a path starting in $(1,1)$ going to $(m,\tilde{m})$, only using steps downwards, to the right and diagonal downwards. A is called monotonic alignment matrix and we denote the set containing all these alignment matrices with $\mathcal{A}(m,\tilde{m})$. The Frobenius inner product $\langle A, C\rangle$ is then the sum of costs along the alignment A. Solving the following minimization problem leads us to a reformulation of the dynamic time warping introduced above as

$$\mathrm{DTW}(C) = \min_{A\in\mathcal{A}(N,M)} \langle A, C\rangle. \tag{9}$$

With Soft DTW we involve all alignments possible in $\mathcal{A}(N,M)$ by replacing the minimization with a *soft minimum*:

$$\min_{x\in S} f(x) \approx \min_{\gamma}{}_{x\in S} f(x) := -\gamma \log \sum_{x\in S} \exp\left(\frac{-f(x)}{\gamma}\right) \tag{10}$$

where S is a discrete subset of the real numbers. This function approximates the minimum of $f(x)$ and is differentiable. The parameter γ controls the tuning between smoothness and approximation of the minimum. Using the DTW-function (9) within (10) yields the expression for Soft Dynamic Time Warping written as

$$\begin{aligned}\mathrm{DTW}_\gamma(\mathbf{x},\tilde{\mathbf{x}}) &= \min_{\gamma}{}_{A\in\mathcal{A}(m,n)} \langle A, C\rangle \\ &= -\gamma \log \sum_{A\in\mathcal{A}(m,n)} \exp\left(\frac{-\langle A, C\rangle}{\gamma}\right).\end{aligned} \tag{11}$$

This is now a differentiable alternative to DTW, which involves all alignments in our cost matrix.

Due to entropic bias[3], Soft DTW can generate negative values, which would cause issues for our use in time series classification. We apply the following remedy to overcome this drawback:

$$\mathrm{Div}(\mathbf{x},\mathbf{y}) = \mathrm{DTW}_\gamma(\mathbf{x},\mathbf{y}) - \frac{1}{2}\cdot\big(\mathrm{DTW}_\gamma(\mathbf{x},\mathbf{x}) + \mathrm{DTW}_\gamma(\mathbf{y},\mathbf{y})\big). \tag{12}$$

This measure is called Soft DTW divergence [43] and will be employed in our experiments.

3.3. Matrix Profile Distance

Another alternative time series measure that has recently been introduced is the *Matrix Profile Distance* (MP distance, [30]). This measure is designed for fast computation and finding similarities between time series.

We will again introduce the concept of the matrix profile of two time series $\mathbf{x}$ and $\tilde{\mathbf{x}}$. The matrix profile is based on the subsequences of these two time series. For a fixed window length L, the subsequence $\mathbf{x}_{i,L}$ of a time series $\mathbf{x}$ is defined as a contiguous L-element subset of $\mathbf{x}$ via $\mathbf{x}_{i,L} = (x_i, x_{i+1}, \ldots, x_{i+L-1})$. The *all-subsequences set* A of $\mathbf{x}$ contains all possible subsequences of $\mathbf{x}$ with length L, $A = \{\mathbf{x}_{1,L}, \mathbf{x}_{2,L}, \ldots, \mathbf{x}_{m-L+1,L}\}$, where m is again the length of $\mathbf{x}$.

For the matrix profile, we need the all-subsequences sets A and B of both time series $\mathbf{x}$ and $\tilde{\mathbf{x}}$. The matrix profile $\mathbf{P}_{\mathrm{ABBA}}$ is the set consisting of the closest Euclidean distances from each

[3] This term is commonly used when the regression results shrink toward a mass at the barycenter of a target [44].

subsequence in A to any subsequence in B and vice versa:

$$\mathbf{P}_{\mathrm{ABBA}} = \left\{ \min_{\tilde{\mathbf{x}}_{j,L} \in B} \|\mathbf{x}_{i,L} - \tilde{\mathbf{x}}_{j,L}\| \;\middle|\; \mathbf{x}_{i,L} \in A \right\} \cup \left\{ \min_{\mathbf{x}_{i,L} \in A} \|\tilde{\mathbf{x}}_{j,L} - \mathbf{x}_{i,L}\| \;\middle|\; \tilde{\mathbf{x}}_{j,L} \in B \right\}$$

With the matrix profile, we can finally define the MP distance based on the idea that two time series are similar if they have many similar subsequences. We do not consider the smallest or the largest value of $\mathbf{P}_{ABBA}$ because then the MP distance could be too rough or too detailed. For example, if we would have two rather similar time series, but either one has a noisy spike or some missing values, then the largest value of the matrix profile could give a wrong impression about the similarity of these two time series. Instead, the distance is defined as

$$\mathrm{MPdist}(X, Y) = k\text{-th smallest value in sorted } \mathbf{P}_{ABBA},$$

where the parameter k is typically set to 5% of $2N$ [30].

We now illustrate the MP distance using an example as illustrated in section 3.3, where we display three time series of length $N = 100$. Our goal is to compare these time series using the MP distance. We observe that X_1 and X_2 have quite similar oscillations. The third time series X_3 does not share any obvious features with the first two sequences.

The MP distance compares the subsequences of the time series, depending on the window length L. Choosing the window length to be $L = 40$, we get the following distances:

$$\mathrm{MPdist}(X_1, X_2) = 0.433,$$
$$\mathrm{MPdist}(X_1, X_3) = 5.425,$$
$$\mathrm{MPdist}(X_2, X_3) = 5.404.$$

As we can see, the MP distance identified the similarity between X_1 and X_2 shows that X_1, X_2 differ from X_3. We also want to show that the MP Distance depends on the window length L. Let us look at the MP distance between the lower oscillation time series X_2 and X_3, which is varying a lot for different values of L as indicated in **Table 1**. Choosing $L = 10$ there is not a large portion of both time series to compare with and as a result we observe a small value for the MP distance, which does not describe the dissimilarity of X_2 and X_3 in a proper way. If we look at $L = 40$, there is a larger part of the time series structure to compare the two series. If there is a special recurring pattern in the time series, the length L should be large enough to cover one recurrence. We illustrate the comparison based on different window lengths in **Figure 4**.

For the tests all data sets consist of time series with a certain length, varying for each data set. Thus we have to decide which window length L should be chosen automatically in the classifier. An empirical study showed that choosing $L \approx N/2$ gives good classification results.

TABLE 1 | MP distance depending on the window length.

L	10	20	30	40
MPdist(X_2, X_3)	0.270	2,034	3,955	5,404

We briefly illustrate the computing times of the different distance measures when applied to time series of increasing length shown in **Figure 5**. It can be seen that DTW is faster than fastDTW. Obviously, the Euclidean distance shows the best scalability. We also observe that the computation of the SDTW is scaling worse than the competing approaches when applied to longer time series.

4. SEMI-SUPERVISED LEARNING BASED ON GRAPH LAPLACIANS

In this section, we focus mainly on two methods that have recently gained wide attention. This first method is inspired by a partial differential equation model originating from material science and the second approach is based on neural networks that incorporate the graph structure of the labeled and unlabeled data.

4.1. Semi-supervised Learning With Phase Field Methods: Allen–Cahn Model

Within the material science community phase field methods have been developed to model the phase separation of a multicomponent alloy system (cf. [45, 46]). The evolution of the phases over time is described by a partial differential equation (PDE) model, such as the Allen-Cahn [46] or Cahn-Hilliard equation [47] both non-linear reaction-diffusion equations of second and fourth order, respectively. These equations can be obtained as gradient flows of the Ginzburg–Landau energy functional

$$\mathcal{E}(u) = \int \frac{\varepsilon}{2} |\nabla u|^2 + \frac{1}{\varepsilon}\phi(u)$$

where u is the order parameter and ε a parameter reflecting the width of the interface between the pure phases. The polynomial ϕ is chosen to have minima at the pure phases, namely $u = -1$ and $u = 1$, to enforce that a minimization of the Ginzburg–Landau energy will lead to phase separation. A common choice is the well-known double-well potential $\phi(u) = \frac{1}{4}(1 - u^2)^2$. The Dirichlet energy term $|\nabla u|^2$ corresponds to minimization of the interfacial length. The minimization is then performed using a gradient flow, which leads to the Allen-Cahn equation

$$u_t = \Delta u - \frac{1}{\varepsilon}\phi'(u) \tag{13}$$

equipped with appropriate boundary and initial conditions. A modified Allen–Cahn equation was used for image inpainting, i.e., restoring damage parts in an image, where a misfit $\omega\left(f - u\right)$ term is added to Equation (13) (cf. [48, 49]). Here, ω is a penalty parameter and f is a function equal to the undamaged image parts or later training data. In [31], Bertozzi and Flenner extended this idea to the case of semi-supervised learning where the training data correspond to the undamaged image parts, i.e, the function

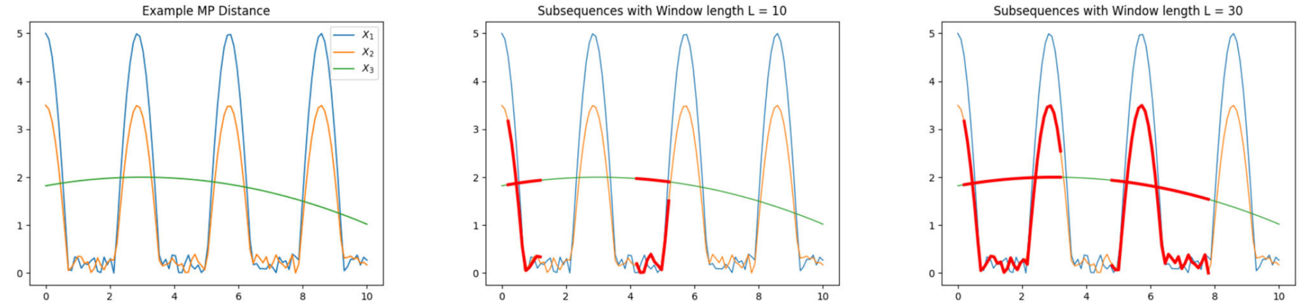


FIGURE 4 | Illustration of Matrix Profile distance **(left)**, subsequences indicated in red with window length $L = 10$ **(middle)** and $L = 30$ **(right)**.

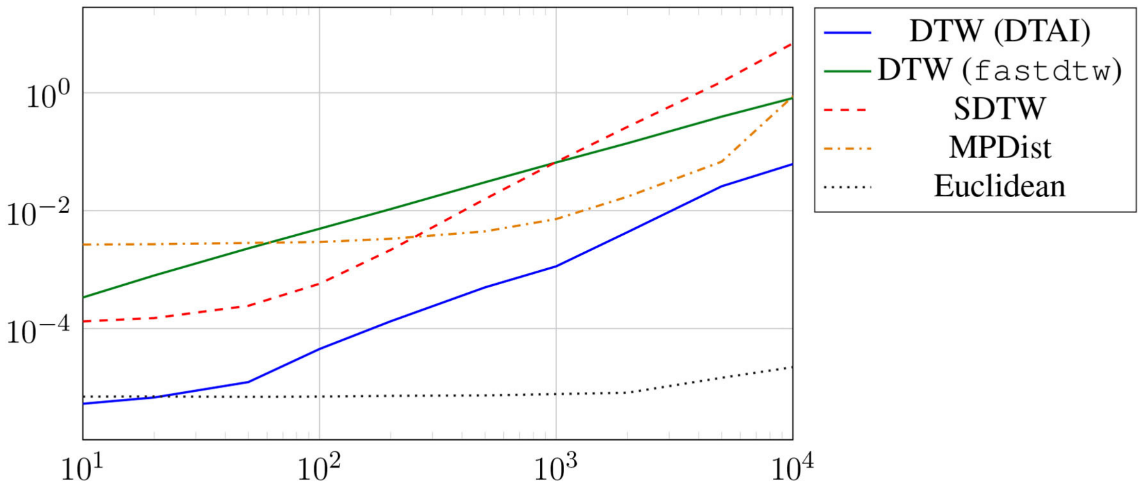


FIGURE 5 | Runtimes of distance computation between a single pair of time series with increasing length.

TABLE 2 | Default parameters used in the experiments.

Method	Parameters and default values
Allen–Cahn	$m_e = 20$, $\varepsilon = \frac{1}{\sqrt{n}}$, $c = \frac{3}{\varepsilon} + \omega$, $\omega = 1e10$, $\tau = 0.01$, $tol = 1e-8$
GCN	10-NN sparsification, $h = 32$, dropout $p = 0.5$, Adam optimization [62], learning rate 0.01, weight decay 0.0005, 500 epochs
Linear System	$\beta = 1$, $tol = 1e-5$
1NN	—

f. Their idea is to consider the modified energy of the following form

$$E(u) = \frac{\varepsilon}{2} u^T L_{\text{sym}} u + \frac{1}{4\varepsilon} \sum_{i=1}^{n} (u_i^2 - 1)^2 + \sum_{i=1}^{n} \frac{\omega_i}{2} (f_i - u_i) \quad (14)$$

where f_i holds the already assigned labels. Here, the first term in (14) reflects the RatioCut based on the graph Laplacian, the second term enforces the pure phases, and the third term corresponds to incorporating the training data. Numerically, this system is solved using a convexity splitting approach [31] where we write

$$E(u) = E_1(u) - E_2(u)$$

with

$$E_1(u) := \frac{\varepsilon}{2} u^T L_{\text{sym}} u + \frac{c}{2} u^T u$$

and

$$E_2(u) := \frac{c}{2} u^T u - \frac{1}{4\varepsilon} + \sum_{i=1}^{n} (u_i^2 - 1)^2 - \sum_{i=1}^{n} \frac{\omega_i}{2} (f_i - u_i)$$

where the positive parameter $c \in \mathbb{R}$ ensures convexity of both energies. In order to compute the minimizer of the above energy we use a gradient scheme where

$$\frac{u^{l+1} - u^l}{\tau} = -\nabla E_1(u^{l+1}) + \nabla E_2(u^l)$$

where the indices k, $k+1$ indicate the current and next time step, respectively. The variable τ is a hyperparameter but can be

TABLE 3 | Study of self-tuning parameters.

		$k = 7$ (%)	**$k = 20$ (%)**	**$k = \sqrt{n}$ (%)**	**$k = 0.1n$ (%)**	**$k = 0.05n$ (%)**
ECG200 ($n = 200$)						
MPDist	GCN	**83.58**	81.74	81.90	81.74	82.54
	Allen-Cahn	**81.00**	79.00	80.00	79.00	80.00
SDTW	GCN	**91.95**	91.34	90.70	91.43	90.55
	Allen-Cahn	**92.00**	90.00	91.00	90.00	91.00
DTW	GCN	88.92	86.76	87.43	86.76	**88.97**
	Allen-Cahn	82.00	82.00	**83.00**	82.00	82.00
SonyAIBORobotSurface1 ($n = 621$)						
MPDist	GCN	**95.45**	88.74	93.08	78.10	89.62
	Allen-Cahn	**75.54**	72.88	73.04	75.37	73.71
SDTW	GCN	90.32	91.46	92.48	87.34	**92.85**
	Allen-Cahn	**93.68**	85.19	82.36	81.36	82.36
DTW	GCN	**97.59**	97.58	97.48	96.49	97.35
	Allen-Cahn	84.03	86.85	87.69	87.19	**88.19**
ECGFiveDays ($n = 884$)						
MPDist	GCN	99.70	**99.77**	99.51	99.66	99.15
	Allen-Cahn	89.89	90.71	95.35	95.82	**96.40**
SDTW	GCN	97.30	97.11	**97.31**	96.49	97.06
	Allen-Cahn	82.00	86.99	85.48	86.76	**87.57**
DTW	GCN	97.22	97.19	**97.39**	97.20	97.35
	Allen-Cahn	**77.35**	76.31	75.72	73.17	74.68
TwoLeadECG *(n = 1,162)*						
MPDist	GCN	**99.81**	99.78	**99.81**	99.62	99.74
	Allen-Cahn	**99.12**	97.10	96.49	97.72	96.57
SDTW	GCN	**92.10**	90.74	90.53	89.98	90.72
	Allen-Cahn	**97.19**	93.24	91.04	87.27	87.71
DTW	GCN	92.94	94.04	94.98	93.97	**96.49**
	Allen-Cahn	93.85	92.36	92.10	**94.12**	93.50

Bold values indicate most accurate classification.

interpreted as a pseudo time-step. In more detail following the notation of [20], this leads to

$$\frac{u^{l+1} - u^l}{\tau} + \varepsilon L_{\text{sym}} u^{l+1} + c u^{l+1} = c u^l - \frac{1}{\varepsilon} \nabla \psi(u^l) + \nabla \phi(u^l)$$

with

$$\psi(u^l) = \sum_{i=1}^{n} ((u_i^l)^2 - 1)^2, \quad \phi(u^l) = \sum_{i=1}^{n} \frac{\omega_i}{2}(f_i - u_i^l).$$

Expanding the order parameter in a number of the small eigenvectors ϕ_i of L_{sym} via $u = \sum_{i=1}^{m_e} a_i \phi_i = \Phi_{m_e} a$ where a is a coefficient vector and $\Phi_{m_e} = [\phi_1, \ldots, \phi_{m_e}]$. This lets us arrive at

$$(1 + \varepsilon \tau \lambda_j a_j^{l+1} + c\tau) a_j^{l+1} = (1 + \tau c) a_j^l - \frac{1}{\varepsilon} b_j^l + d_j^l, \quad \forall j = 1, \ldots, m_e$$

using

$$b^l = \Phi_{m_e}^T \nabla \psi(\Phi_{m_e} a^l), \quad d^l = \Phi_{m_e}^T \nabla \phi(\Phi_{m_e} a^l).$$

In [50], the authors extend this to the case of multiple classes where again the spectral information of the graph Laplacian are crucial as the energy term includes $\frac{\varepsilon}{2} \operatorname{tr}(U^T L_{\text{sym}} U)$ with $U \in \mathbb{R}^{n,s}$, s being the number of classes for segmentation, and tr being the trace of the matrix. Details of the definition of the potential and the fidelity term incorporating the training data are found in [50]. Further extensions of this approach have been suggested in [20, 22, 51–55].

4.2. Semi-supervised Learning Based on Graph Convolutional Networks

Artificial neural networks and in particular deep neural networks have shown outstanding performance in many learning tasks [56, 57]. The incorporation of additional structural information via a graph structure has received wide attention [24] with particular success within the semi-supervised learning formulation [21].

Let $\mathbf{h}_i^{(l)}$ denote the hidden feature vector of the i-th node in the l-th layer. The feature mapping of a simple multilayer perceptron (MLP) computes the new features by multiplying with a weight matrix $\Theta^{(l)T}$ and adding a bias vector $\mathbf{b}^{(l)}$, then applying a (potentially layer-dependent) ReLU activation function σ_l in all

TABLE 4 | Varying the number of eigenpairs for the reduced Allen–Cahn equation.

Number of eigenvalues	**10 (%)**	**20 (%)**	**30 (%)**	**150 (%)**	**190 (%)**
Dataset ECG200					
MPDist	82.00	81.00	**86.00**	62.00	56.00
SDTW	78.00	**92.00**	**92.00**	68.00	66.00
DTW	78.00	82.00	**87.00**	69.00	54.00
Number of eigenvalues	**10 (%)**	**20 (%)**	**30 (%)**	**500 (%)**	**600 (%)**
SonyAIBORobotSurface1					
MPDist	**85.36**	75.54	73.04	51.58	51.08
SDTW	**96.17**	93.68	83.19	52.08	49.92
DTW	**90.01**	84.03	72.71	52.41	48.58
Number of eigenvalues	**10 (%)**	**20 (%)**	**30 (%)**	**700 (%)**	**800 (%)**
ECGFiveDays					
MPDist	87.19	**89.89**	85.95	50.29	51.22
SDTW	**91.52**	82.00	84.20	54.00	52.38
DTW	68.87	**77.35**	77.00	49.82	50.29

Bold values indicate most accurate classification.

layers except the last. This layer operation can be written as $\mathbf{h}_i^l = \sigma_l\Big(\Theta^{(l)T}\mathbf{h}_i^{(l-1)} + \mathbf{b}^{(l)}\Big)$.

In Graph Neural Networks, the features are additionally propagated along the edges of the graph. This is achieved by forming weighted sums over the local neighborhood of each node, leading to

$$\mathbf{h}_i^l = \sigma_l\Big(\sum_{j\in\mathcal{N}_i\cup\{i\}} \frac{\hat{w}_{ij}}{\sqrt{\hat{d}_i\hat{d}_j}}\Theta^{(l)T}\mathbf{h}_j^{(l-1)} + \mathbf{b}^{(l)}\Big). \tag{15}$$

Here, $\mathcal{N}_i$ denotes the set of neighbors of node i, $\Theta^{(l)}$ and $\mathbf{b}^{(l)}$ the trainable parameters of layer l, the $\hat{w}_{ij}$ denote the entries of the adjacency matrix W with added self loops, $\hat{W} = W + I$, and the $\hat{d}_i$ denote the row sums of that matrix. By adding the self loops, it is ensured that the original features of that node are maintained in the weighted sum.

To obtain a matrix formulation, we can accumulate state matrices $X^{(l)}$ whose n rows are the feature vectors $\mathbf{h}_i^{(l)T}$ for $i = 1, \ldots, n$. The propagation scheme of a simple two-layer graph convolutional network can then be written as

$$\begin{aligned} X^{(1)} &= \sigma\Big(\hat{D}^{-1/2}\hat{W}\hat{D}^{-1/2}X^{(0)}\Theta^{(1)} + \mathbf{b}^{(1)}\Big) \\ X^{(2)} &= \quad \hat{D}^{-1/2}\hat{W}\hat{D}^{-1/2}X^{(1)}\Theta^{(2)} + \mathbf{b}^{(2)} \end{aligned} \tag{16}$$

where $\hat{D}$ is the diagonal matrix holding the $\hat{d}_i$.

Multiplication with $\hat{D}^{-1/2}\hat{W}\hat{D}^{-1/2}$ can also be understood in a spectral sense as performing *graph convolution* with the spectral filter function $\varphi(\lambda) = 1 - \lambda$. This filter originates from truncating a Chebyshev polynomial to first order as discussed in [58]. As a result of this filter the eigenvalues λ of the graph Laplacian operator $\mathcal{L}$ (formed in this case *after* adding the self loops) are transformed via φ to obtain damping coefficients for the corresponding eigenvectors. This filter has been shown to lead to convolutional layers equivalent to aggregating node representations from their direct neighborhood (cf. [58] for more information).

It has been noted, e.g., in [59] that traditional graph neural networks including GCN are mostly targeted at the case of *sparse* graphs, where each node is only connected to a small number of neighbors. The fully connected graphs that we utilize in this work present challenges for GCN through their spectral properties. Most notably, these *dense* graphs typically have large eigengaps, i.e., the gap between the smallest eigenvalue $\lambda_1 = 0$ and the second eigenvalue $\lambda_2 > 0$ may be close to 1. Hence the GCN filter acts almost like a projection onto the undesirable eigenvector ϕ_1. However, it has been observed in the same work that in some applications, GCNs applied to *sparsified* graphs yield comparable results to dedicated dense methods. Our experiments justified only using Standard GCN on a k-nearest neighbor subgraph.

4.3. Other Semi-supervised Learning Methods

In the context of graph-based semi-supervised learning a rather straightforward approach follows from minimizing the following objective

$$\min_u \frac{1}{2}\left\|u - f\right\|_2^2 + \frac{\beta}{2}u^T L_{\text{sym}} u \tag{17}$$

where f holds the values 1, -1, and 0 according to the labeled and unlabeled data. Calculating the derivative shows that in order to obtain u, we need to solve the following *linear system* of equations

$$\left(I + \beta L_{\text{sym}}\right) u = f$$

where I is the identity matrix of the appropriate dimensionality.

Furthermore, we compare our previously introduced approaches to the well-known one-nearest neighbor (1NN) method. In the context of time series classification this method was proposed in [11]. In each iteration, we identify the indices i, j with the shortest distance between the labeled sample $\mathbf{x}_i$ and the unlabeled sample $\mathbf{x}_j$. The label of $\mathbf{x}_i$ is then copied to $\mathbf{x}_j$. This process is repeated until no unlabeled data remain.

In [60], the authors construct several graph Laplacians and then perform the semi-supervised learning based on a weighted sum of the Laplacian matrices.

5. NUMERICAL EXPERIMENTS

In this section, we illustrate how the algorithms discussed in this paper perform when applied to multiple time series data sets. We here focus on binary classification and use time series taken from the UCR time series classification archive [4] [61]. All our codes are to be found at https://github.com/dominikalfke/TimeSeriesSSL. The distance measure we use here are the previously introduced

[4] We focussed on all binary classification series listed in `TwoClassProblems.csv` within http://www.timeseriesclassification.com/Downloads/Archives/Univariate2018_arff.zip.

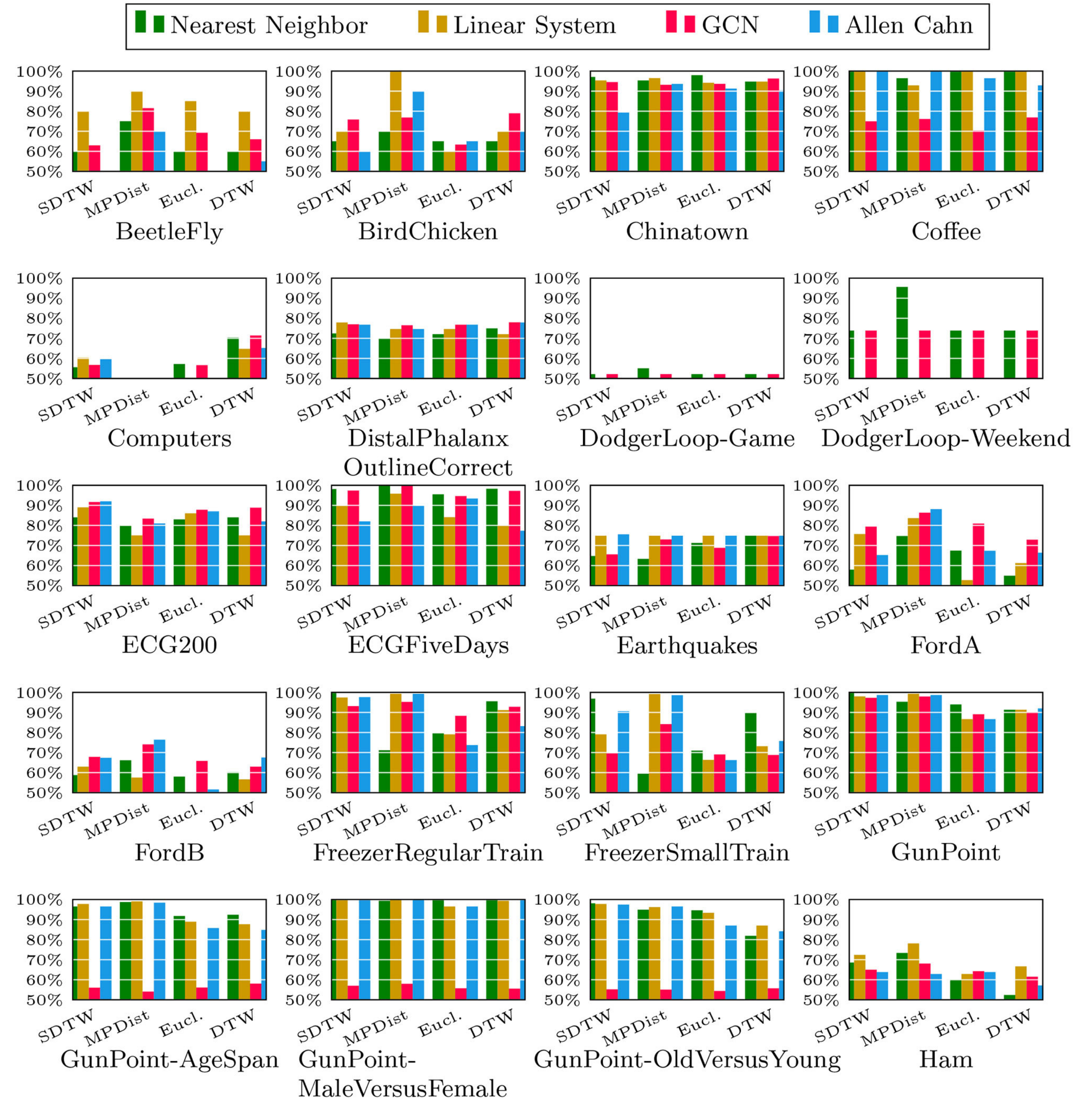


FIGURE 6 | Comparison of the proposed methods using various distance measures for a variety of time series data. The size of the training set is specified in `TwoClassProblems.csv` within http://www.timeseriesclassification.com/Downloads/Archives/Univariate2018_arff.zip.

DTW, Soft DTW divergence, MP, and Euclidean distances. For completeness, we list the default parameters for all methods in **Table 2**.

We split the presentation of the numerical results in the following way. We start by exploring the dependence of our schemes on some of the hyperparameters inherent in their derivation. We start by investigating the self-tuning parameters, namely the value of the chosen neighbor to compute the local scaling. We then study the performance of the Allen–Cahn model depending on the number of eigenpairs used for the approximation of the graph Laplacian. For our main study, we pair up all distance measures with all learning methods and report the results on all datasets. Furthermore, we investigate how the method's performance depends on the number of available training data using random training splits.

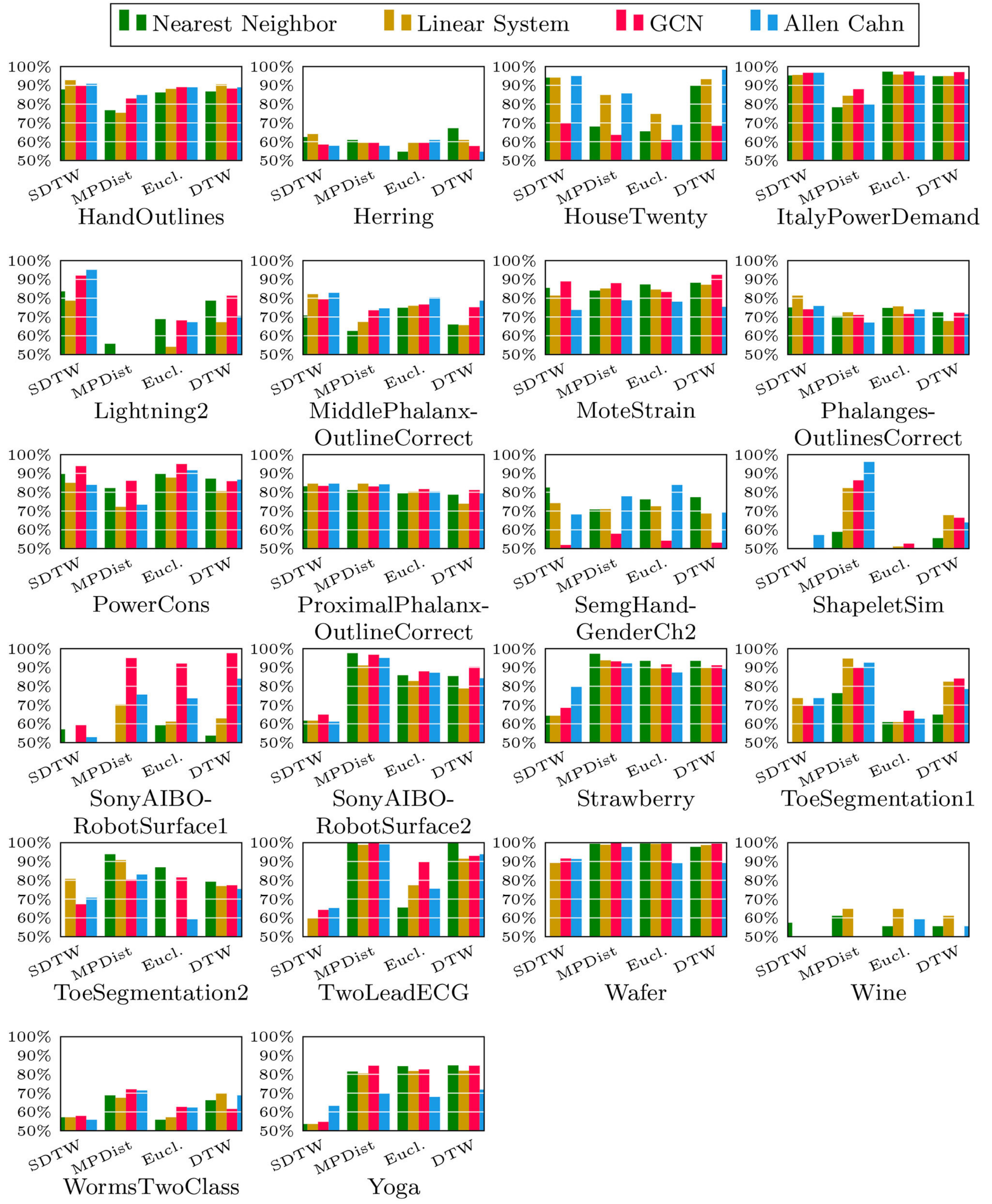


FIGURE 7 | Comparison of the proposed methods using various distance measures for a variety of time series data.

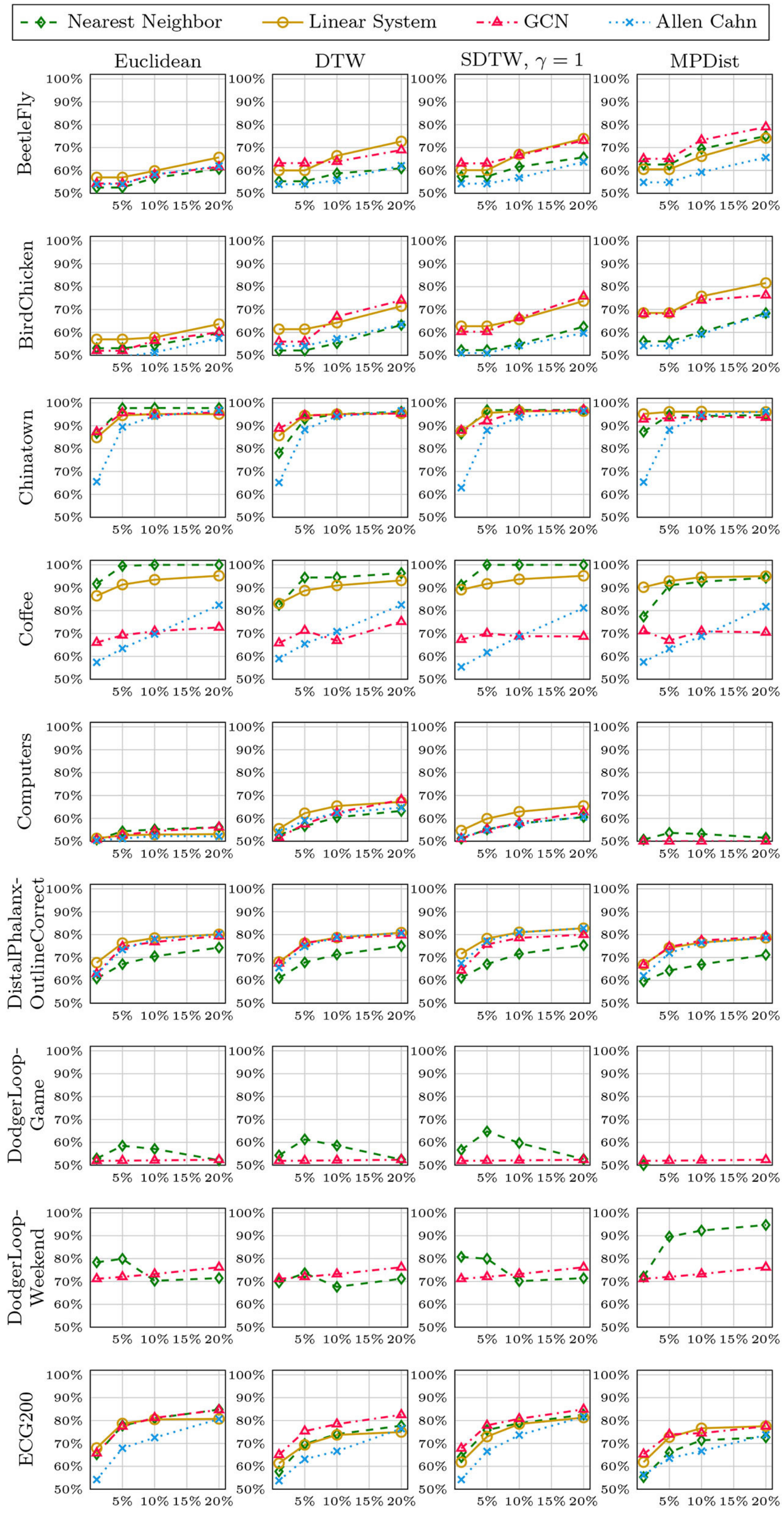


FIGURE 8 | Method accuracy comparison for random training splits of different sizes (part 1/5).

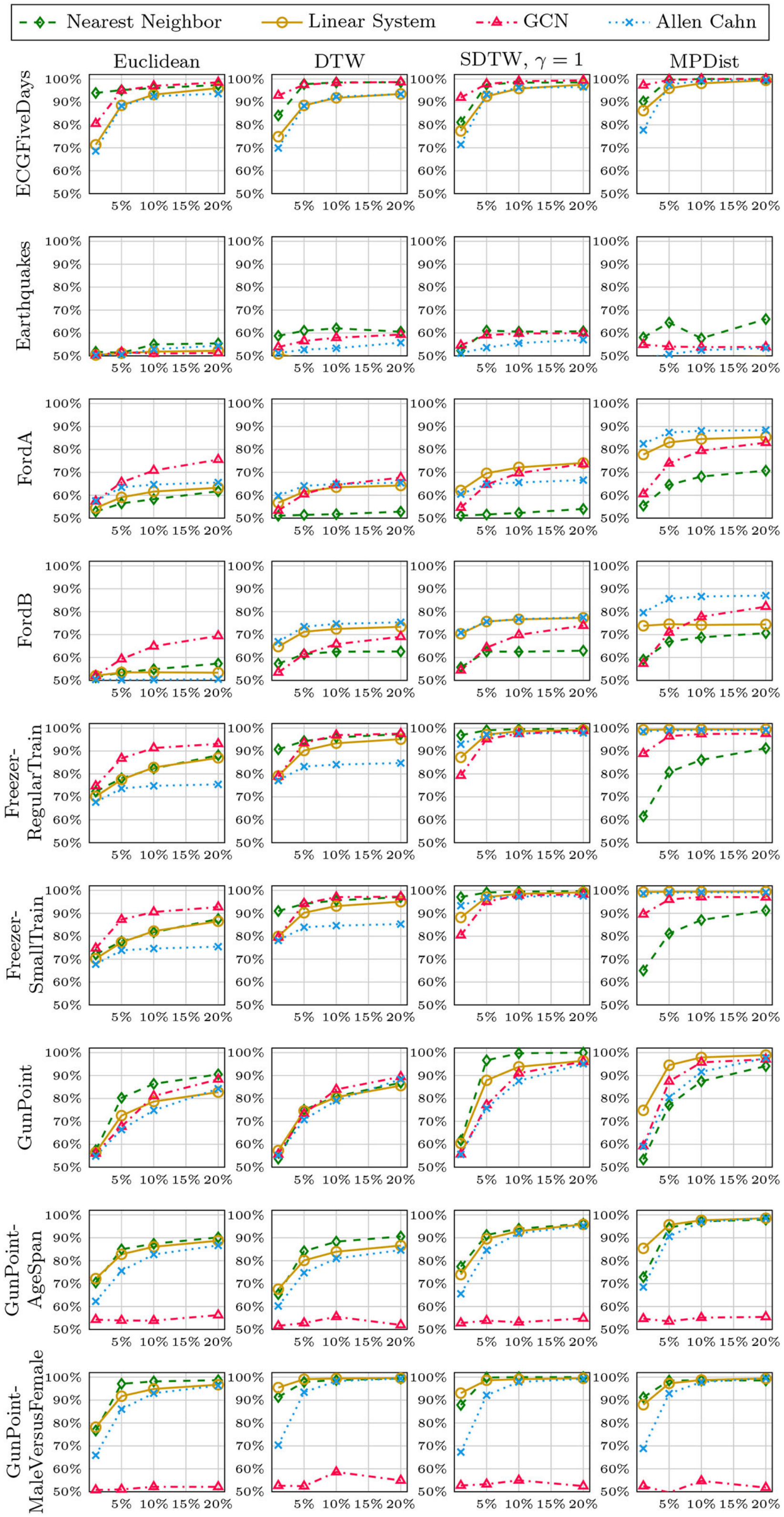


FIGURE 9 | Method accuracy comparison for random training splits of different sizes (part 2/5).

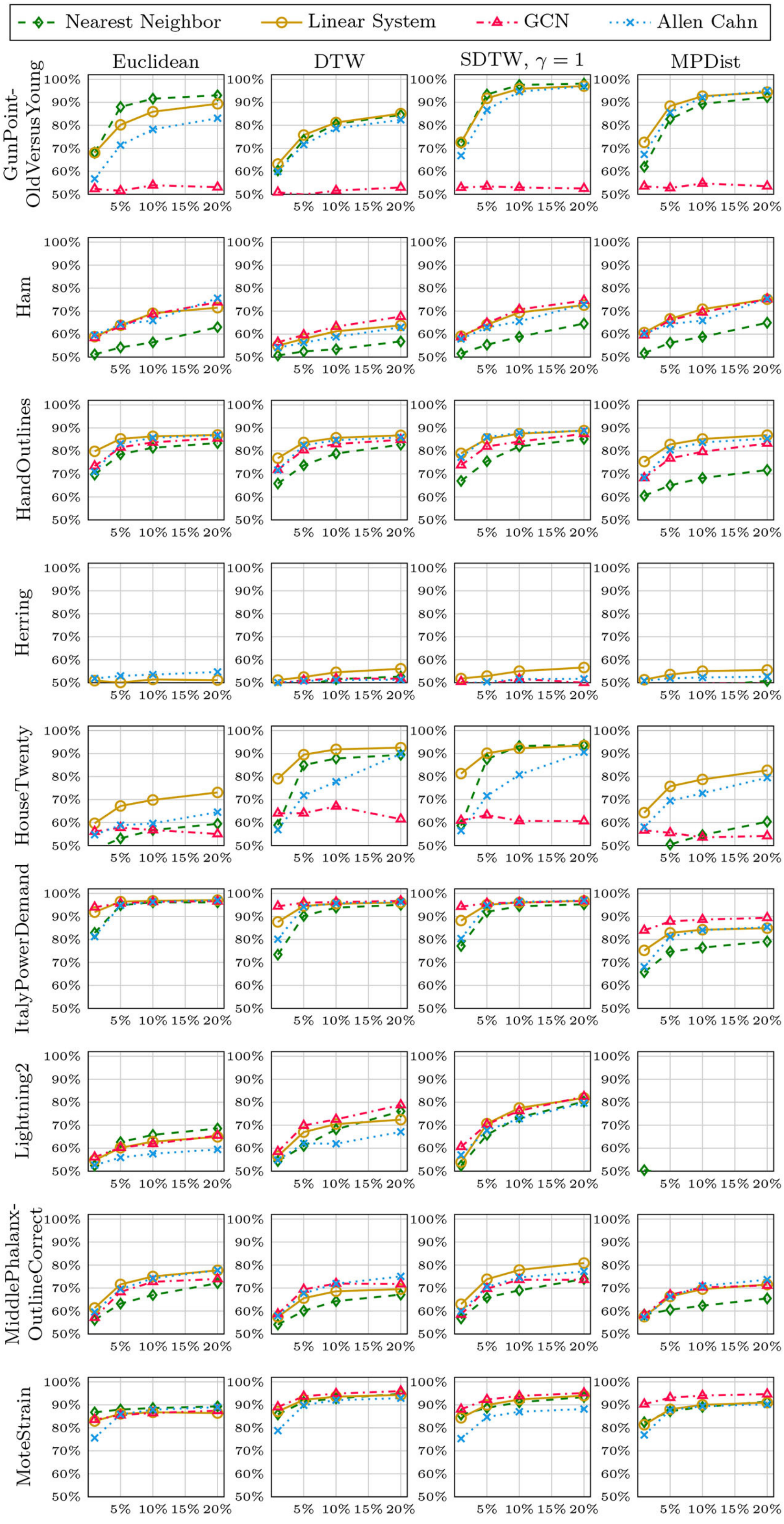


FIGURE 10 | Method accuracy comparison for random training splits of different sizes (part 3/5).

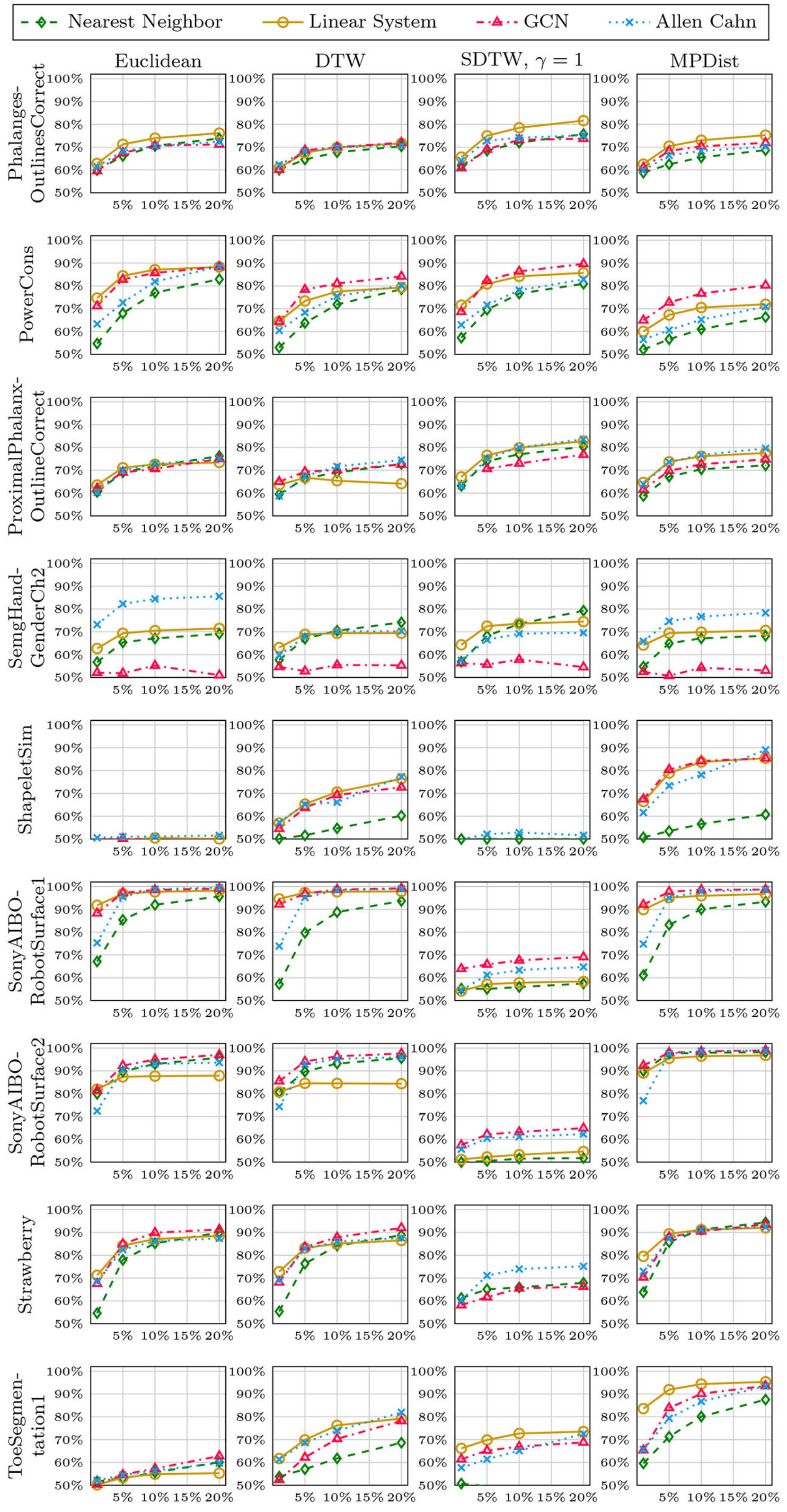


FIGURE 11 | Method accuracy comparison for random training splits of different sizes (part 4/5).

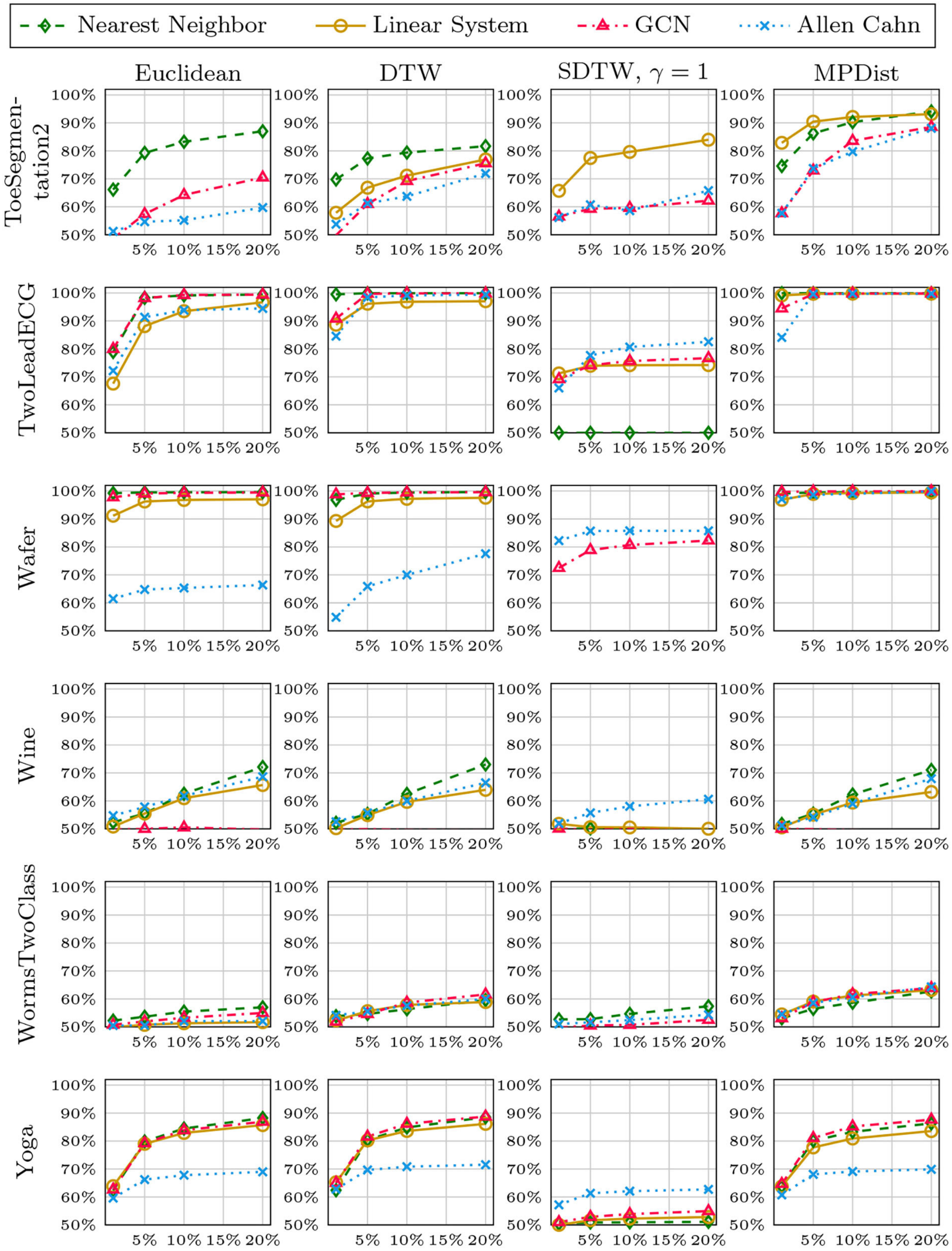


FIGURE 12 | Method accuracy comparison for random training splits of different sizes (part 5/5).

5.1. Self-Tuning Values

In section 2, we proposed the use of the self-tuning approach for the Gaussian function within the weight matrix. The crucial hyperparameter we want to explore now is the choice of neighbor k for the construction of $\sigma_i = \text{dist}(\mathbf{x}_i, \mathbf{x}_{k,i})$ with $\mathbf{x}_{k,i}$ the k-th nearest neighbor of the data point $\mathbf{x}_i$. We can see from **Table 3** that the small values $k = 7, 20$ perform quite well in comparison to the larger self-tuning parameters. As a result we will use these smaller values in all further computations.

5.2. Spectral Approximation

As described in section 4 the Allen–Cahn equation is projected to a lower-dimensional space using the insightful information provided by the eigenvectors to the smallest eigenvalues of the graph Laplacian. We now investigate how the number of used eigenvectors impacts the accuracy. In the following we vary the number of eigenvalues from 10 to 190 and compare the performance of the Allen–Cahn method on three different datasets. The results are shown in **Table 4** and it becomes clear that a vast number of eigenvectors does not lead to better classification accuracy. As a result we require a smaller number of eigenpair computations and also fewer computations within the Allen–Cahn scheme itself. The comparison was done for the self-tuning parameter $k = 7$.

5.3. Full Method Comparison

We now compare the Allen-Cahn approach, the GCN scheme, the linear systems based method, and the 1NN algorithm, each paired up with each of the distance measures introduced in section 3. Full results are listed in **Figures 6**, **7**. We show the comparison for all 42 datasets.

As can be seen there are several datasets where the performance of all methods is fairly similar even when the distance measure is varied. Here, we name `Chinatown`, `Earthquakes`, `GunPoint`, `ItalyPowerDemand`, `MoteStrain`, `Wafer`. There are several examples where the methods do not seem to perform well, with GCN and 1NN relatively similar outperforming the Linear System and Allen–Cahn approach. Such examples are `DodgerLoopGame`, `DodgerLoopWeekend`. The GCN method clearly does not perform well with the GunPoint datasets where the other methods clearly perform well. It is surprising to note that the Euclidean distance, given its computational speed and simplicity, does not come out as underperforming with respect to the accuracy across the different methods. There are very few datasets where one distance clearly outperforms the other choice. We name `ShapeletSim`, `ToeSegementation1` here. One might conjecture that the varying sizes of the training data might be a reason for the difference in performance of the models. To investigate this further we will next vary the training splits for all datasets and methods.

5.4. Varying Training Splits

In **Figures 8–12**, we vary the size of the training set from 1 to 20% of the available data. All reported numbers are averages over 100 random splits. The numbers we observe mirror the performance of the full training size. We see that the methods show reduced performance when only 1% of the training data are used but often reach an accuracy plateau when 5 to 10% of the training data are used. We observe that the size of the training set alone does not explain the different performance in the various datasets and methods applied here.

6. CONCLUSION

In this paper we took to the task of classifying time series data in a semi-supervised learning setting. For this we proposed to represent the data as a fully-connected graph where the edge weights are created based on a Gaussian similarity measure (1). The heart of this function is the difference measure between the time series, for which we used the (Soft) Dynamic Time Warping and Matrix Profile based distance measures as well as the Euclidean distance. We then investigated several learning algorithms, namely, the Allen–Cahn-based method, the Graph Convolutional Network scheme, and a linear system approach, all reliant on the graph Laplacian, as well as the Nearest Neighbor method. We then illustrated the performance of all pairs of distance measure and learning methods. In this empirical study we observed that the methods tend to show an increased performance adding more training data. Studying all binary time-series with the timeseriesclassification.com repository gives results that in accordance with the no free lunch theorem show no clear winner. On the positive side the methods often perform quite well and there are only a few datasets with decreased performance. The comparison of the distance measures indicates there are certain cases where they outperform their competitors but also there is no clear winner with regards to accuracy. We believe that this empirical, reproducible study will encourage further research in this direction. Additionally, it might be interesting to consider model-based representations of time-series such as ARMA [63, 64] to use within the graph representations used here.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

MG provide the initial implementation of some of the methods. MS supervised the other members, wrote parts of the manuscript, and implemented the Allen Cahn scheme. DB implemented the GCN approach, wrote parts of the manuscript, and oversaw the design of the tests. LP implemented several algorithms and wrote parts of the manuscript. All authors contributed to the article and approved the submitted version.

FUNDING

MS and LP acknowledge the funding of the BMBF grant 01|S20053A. DB was partially supported by KINTUC project [Sächsische Aufbaubank–Förderbank–(SAB) 100378180].

ACKNOWLEDGMENTS

All authors would like to acknowledge the hard work and dedication by the team maintaining www.timeseriesclassification.com/. The publication of this article was funded by Chemnitz University of Technology.

REFERENCES

1. Fu TC. A review on time series data mining. *Eng Appl Artif Intell*. (2011) 24:164–81. doi: 10.1016/j.engappai.2010.09.007
2. Bello-Orgaz G, Jung JJ, Camacho D. Social big data: recent achievements and new challenges. *Inform Fusion*. (2016) 28:45–59. doi: 10.1016/j.inffus.2015.08.005
3. Chen F, Deng P, Wan J, Zhang D, Vasilakos AV, Rong X. Data mining for the internet of things: literature review and challenges. *Int J Distribut Sensor Netw*. (2015) 11:431047. doi: 10.1155/2015/431047
4. Laptev N, Amizadeh S, Flint I. Generic and scalable framework for automated time-series anomaly detection. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (2015). p. 1939–47. doi: 10.1145/2783258.2788611
5. Chiu B, Keogh E, Lonardi S. Probabilistic discovery of time series motifs. In: *Proceedings of the ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (2003). p. 493–8. doi: 10.1145/956750.956808
6. De Gooijer JG, Hyndman RJ. 25 years of time series forecasting. *Int J Forecast* (2006). 22:443–73. doi: 10.1016/j.ijforecast.2006.01.001
7. Wei WW. Time series analysis. In: Todd little editor. *The Oxford Handbook of Quantitative Methods in Psychology*. Vol. 2. (2006).
8. Chatfield C, Xing H. *The Analysis of Time Series: An Introduction with R*. CRC Press. (2019) doi: 10.1201/9781351259446
9. Fawaz HI, Forestier G, Weber J, Idoumghar L, Muller PA. Deep learning for time series classification: a review. *Data Mining Knowledge Discov*. (2019) 33:917–63. doi: 10.1007/s10618-019-00619-1
10. Abanda A, Mori U, Lozano JA. A review on distance based time series classification. *Data Mining Knowledge Discov*. (2019) 33:378–412. doi: 10.1007/s10618-018-0596-4
11. Wei L, Keogh E. Semi-supervised time series classification. In: *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (2006). p. 748–53. doi: 10.1145/1150402.1150498
12. Liao TW. Clustering of time series data-a survey. *Pattern Recogn*. (2005) 38:1857–74. doi: 10.1016/j.patcog.2005.01.025
13. Aghabozorgi S, Shirkhorshidi AS, Wah TY. Time-series clustering-a decade review. *Inform Syst*. (2015) 53:16–38. doi: 10.1016/j.is.2015.04.007
14. Shifaz A, Pelletier C, Petitjean F, Webb GI. TS-CHIEF: a scalable and accurate forest algorithm for time series classification. *Data Mining Knowledge Discov*. (2020) 34:742–75. doi: 10.1007/s10618-020-00679-8
15. Dempster A, Petitjean F, Webb GI. ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining Knowledge Discov*. (2020) 34:1454–95. doi: 10.1007/s10618-020-00701-z
16. Fawaz HI, Lucas B, Forestier G, Pelletier C, Schmidt DF, Weber J, et al. Inceptiontime: Finding alexnet for time series classification. *Data Mining Knowledge Discov*. (2020) 34:1936–62. doi: 10.1007/s10618-020-00710-y
17. Zhu X, Goldberg A. *Introduction to Semi-supervised Learning*. Morgan & Claypool Publishers (2009). doi: 10.2200/S00196ED1V01Y200906AIM006
18. Chapelle O, Schölkopf B, Zien A. Semi-supervised learning. *IEEE Trans Neural Netw*. (2009) 20:542. doi: 10.1109/TNN.2009.2015974
19. Stoll M. A literature survey of matrix methods for data science. *GAMM-Mitt*. (2020) 43:e202000013:4. doi: 10.1002/gamm.202000013
20. Mercado P, Bosch J, Stoll M. Node classification for signed social networks using diffuse interface methods. In: *ECMLPKDD*. (2019). doi: 10.1007/978-3-030-46150-8_31
21. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. *arXiv [Preprint]. arXiv:160902907* (2016).
22. Bertozzi AL, Luo X, Stuart AM, Zygalakis KC. Uncertainty quantification in graph-based classification of high dimensional data. *SIAM/ASA J Uncertainty Quant*. (2018) 6:568–95. doi: 10.1137/17M1134214
23. von Luxburg U. A tutorial on spectral clustering. *Stat Comput*. (2007) 17:395–416. doi: 10.1007/s11222-007-9033-z
24. Bruna J, Zaremba W, Szlam A, LeCun Y. Spectral networks and locally connected networks on graphs. *arXiv [Preprint]. arXiv:13126203* (2013).
25. Chung FR, Graham FC. *Spectral graph Theory*. American Mathematical Soc. (1997).
26. Jung A. Networked exponential families for big data over networks. *IEEE Access*. (2020) 8:202897–909. doi: 10.1109/ACCESS.2020.3033817
27. Jung A, Tran N. Localized linear regression in networked data. *IEEE Signal Process Lett*. (2019) 26:1090–4. doi: 10.1109/LSP.2019.2918933
28. Müller M. *Information Retrieval for Music and Motion*. vol. 2. Springer (2007). doi: 10.1007/978-3-540-74048-3
29. Cuturi M, Blondel M. Soft-DTW: a differentiable loss function for time-series. In: *International Conference on Machine Learning*. PMLR (2017). p. 894–903.
30. Gharghabi S, Imani S, Bagnall A, Darvishzadeh A, Keogh E. An ultra-fast time series distance measure to allow data mining in more complex real-world deployments. *Data Mining Knowledge Discov*. (2020) 34:1104–35. doi: 10.1007/s10618-020-00695-8
31. Bertozzi AL, Flenner A. Diffuse interface models on graphs for classification of high dimensional data. *Multiscale Model Simul*. (2012) 10:1090–118. doi: 10.1137/11083109X
32. Bagnall A, Lines J, Bostrom A, Large J, Keogh E. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data mining Knowledge Discov*. (2017) 31:606–60. doi: 10.1007/s10618-016-0483-9
33. Ruiz AP, Flynn M, Large J, Middlehurst M, Bagnall A. The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining Knowledge Discov*. (2021) 35:401–49. doi: 10.1007/s10618-020-00727-3
34. MacQueen J. Some methods for classification and analysis of multivariate observations. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. Oakland, CA (1967). p. 281–97.
35. MacKay DJ, Mac Kay DJ. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press (2003).
36. Belkin M, Niyogi P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In: *Advances in Neural Information Processing Systems*. (2001). p. 585–91.
37. Shawe-Taylor J, Cristianini N. *Kernel Methods for Pattern Analysis*. Cambridge University Press (2004). doi: 10.1017/CBO9780511809682
38. Hofmann T, Schölkopf B, Smola AJ. Kernel methods in machine learning. *Ann Stat*. (2008) 36:1171–220. doi: 10.1214/009053607000000677
39. Zelnik-Manor L, Perona P. Self-tuning spectral clustering. In: *Advances in Neural Information Processing Systems*. (2005). p. 1601–8.
40. Keogh E, Kasetty S. On the need for time series data mining benchmarks: a survey and empirical demonstration. *Data Mining Knowledge Discov*. (2003) 7:349–71. doi: 10.1023/A:1024988512476
41. Salvador S, Chan PK. Toward accurate dynamic time warping in linear time and space. *Intell Data Anal*. (2004) 11:70–80. doi: 10.3233/IDA-2007-11508
42. Wu R, Keogh EJ. FastDTW is approximate and generally slower than the algorithm it approximates. *IEEE Trans Knowledge Data Eng*. (2020). doi: 10.1109/TKDE.2020.3033752
43. Blondel M, Mensch A, Vert JP. Differentiable divergences between time series. *arXiv [Preprint]. arXiv:201008354* (2020).

44. Lin H, Hong X, Ma Z, Wei X, Qiu Y, Wang Y, et al. Direct measure matching for crowd counting. *arXiv [Preprint]. arXiv:210701558* (2021). doi: 10.24963/ijcai.2021/116
45. Taylor JE, Cahn JW. Linking anisotropic sharp and diffuse surface motion laws via gradient flows. *J Statist Phys*. (1994) 77:183–97. doi: 10.1007/BF02186838
46. Allen SM, Cahn JW. A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening. *Acta Metall*. (1979) 27:1085–95. doi: 10.1016/0001-6160(79)90196-2
47. Cahn JW, Hilliard JE. Free energy of a nonuniform system. I. Interfacial free energy. *J Chem Phys*. (1958) 28:258–67. doi: 10.1063/1.1744102
48. Bosch J, Kay D, Stoll M, Wathen A. Fast solvers for Cahn-Hilliard inpainting. *SIAM J Imaging Sci*. (2014) 7:67–97. doi: 10.1137/130921842
49. Bertozzi AL, Esedoglu S, Gillette A. Inpainting of binary images using the Cahn-Hilliard equation. *IEEE Trans Image Process*. (2007) 16:285–91. doi: 10.1109/TIP.2006.887728
50. Garcia-Cardona C, Merkurjev E, Bertozzi AL, Flenner A, Percus AG. Multiclass data segmentation using diffuse interface methods on graphs. *IEEE Trans Pattern Anal Mach Intell*. (2014) 36:1600–13. doi: 10.1109/TPAMI.2014.2300478
51. Bosch J, Klamt S, Stoll M. Generalizing diffuse interface methods on graphs: nonsmooth potentials and hypergraphs. *SIAM J Appl Math*. (2018) 78:1350–77. doi: 10.1137/17M1117835
52. Bergermann K, Stoll M, Volkmer T. Semi-supervised learning for multilayer graphs using diffuse interface methods and fast matrix vector products. *SIAM J Math Data Sci*. (2021). doi: 10.1137/20M1352028
53. Budd J, van Gennip Y. Graph MBO as a semi-discrete implicit Euler scheme for graph Allen-Cahn. *arXiv [Preprint]. arXiv:190710774* (2019). doi: 10.1137/19M1277394
54. Budd J, van Gennip Y, Latz J. Classification and image processing with a semi-discrete scheme for fidelity forced Allen-Cahn on graphs. *arXiv [Preprint]. arXiv:201014556* (2020). doi: 10.1002/gamm.202100004
55. Calatroni L, van Gennip Y, Schönlieb CB, Rowland HM, Flenner A. Graph clustering, variational image segmentation methods and Hough transform scale detection for object measurement in images. *J Math Imaging Vision*. (2017) 57:269–91. doi: 10.1007/s10851-016-0678-0
56. Goodfellow I, Bengio Y, Courville A, Bengio Y. *Deep Learning*. vol. 1. Cambridge: MIT Press (2016).
57. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*. (2015) 521:436–44. doi: 10.1038/nature14539
58. Zhang S, Tong H, Xu J, Maciejewski R. Graph convolutional networks: a comprehensive review. *Comput Soc Netw*. (2019) 6:1–23. doi: 10.1186/s40649-019-0069-y
59. Alfke D, Stoll M. Pseudoinverse graph convolutional networks: fast filters tailored for large eigengaps of dense graphs and hypergraphs. *Data Mining Knowledge Discov*. (2021). doi: 10.1007/s10618-021-00752-w
60. Xu Z, Funaya K. Time series analysis with graph-based semi-supervised learning. In: *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE (2015). p. 1–6. doi: 10.1109/DSAA.2015.7344902
61. Dau HA, Bagnall A, Kamgar K, Yeh CCM, Zhu Y, Gharghabi S, et al. The UCR time series archive. *IEEE/CAA J Automat Sin*. (2019) 6:1293–305. doi: 10.1109/JAS.2019.1911747
62. Kingma D, Ba JL. Adam: a method for stochastic optimization. In: *Proc Int Conf Learn Represent. ICLR'15*. (2015).
63. Brockwell PJ, Davis RA. *Time Series: Theory and Methods*. Springer Science & Business Media (2009).
64. Spiegel S. *Time series distance measures*. Ph.D. thesis. Berlin, Germany.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Applied Mathematics and Statistics

ORIGINAL RESEARCH
published: 26 January 2022
doi: 10.3389/fams.2021.780322

A New Two-Parameter Estimator for Beta Regression Model: Method, Simulation, and Application

Mohamed R. Abonazel [1]*, *Zakariya Yahya Algamal* [2], *Fuad A. Awwad* [3] *and Ibrahim M. Taha* [4]

[1] Department of Applied Statistics and Econometrics, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza, Egypt, [2] Department of Statistics and Informatics, University of Mosul, Mosul, Iraq, [3] Department of Quantitative Analysis, College of Business Administration, King Saud University, Riyadh, Saudi Arabia, [4] Department of Mathematics, Statistics, and Insurance, Sadat Academy for Management Sciences, Tanta, Egypt

OPEN ACCESS

Edited by:
Gareth William Peters,
University of California, Santa Barbara,
United States

Reviewed by:
Mingjun Zhong,
University of Aberdeen,
United Kingdom
Mohammad Arashi,
Ferdowsi University of Mashhad, Iran

Correspondence:
Mohamed R. Abonazel
mabonazel@cu.edu.eg

Specialty section:
This article was submitted to Statistics,
a section of the journal
Frontiers in Applied Mathematics and Statistics

Received: *20 September 2021*
Accepted: *17 December 2021*
Published: *26 January 2022*

Citation:
Abonazel MR, Algamal ZY, Awwad FA and Taha IM (2022) A New Two-Parameter Estimator for Beta Regression Model: Method, Simulation, and Application. Front. Appl. Math. Stat. 7:780322. doi: 10.3389/fams.2021.780322

The beta regression is a widely known statistical model when the response (or the dependent) variable has the form of fractions or percentages. In most of the situations in beta regression, the explanatory variables are related to each other which is commonly known as the multicollinearity problem. It is well-known that the multicollinearity problem affects severely the variance of maximum likelihood (ML) estimates. In this article, we developed a new biased estimator (called a two-parameter estimator) for the beta regression model to handle this problem and decrease the variance of the estimation. The properties of the proposed estimator are derived. Furthermore, the performance of the proposed estimator is compared with the ML estimator and other common biased (ridge, Liu, and Liu-type) estimators depending on the mean squared error criterion by making a Monte Carlo simulation study and through two real data applications. The results of the simulation and applications indicated that the proposed estimator outperformed ML, ridge, Liu, and Liu-type estimators.

Keywords: biased estimation, Fisher's scoring, mean squared error (MSE), multicollinearity, Liu beta regression, relative efficiency, ridge beta regression, two-parameter estimator

INTRODUCTION

The beta regression model has been common in many areas, primarily economic and medical research, such as income share, unemployment rates in certain nations, the Gini index for each region, graduation rates in major universities, or the percentage of body fat in medical subjects. Beta regression model, such as any regression model in the context of generalized linear models (GLMs) is used to examine the effect of certain explanatory variables on a non-normal response variable. However, in the case of beta regression, the response component is restricted to an interval (0, 1), such as proportions, percentages, and fractions.

Multicollinearity is a popular issue in econometric modeling. It indicates that there is a strong association between the explanatory variables. It is well-established that the covariance matrix of the maximum likelihood (ML) estimator is ill-conditioned in the case of severing multicollinearity. One of the negative consequences of this issue is that the variance of the regression coefficients gets inflated. As a consequence, the significance and the magnitude of the coefficients are affected. Many of the conventional approaches used to address this issue include: gathering additional data, re-specifying the model, or removing the correlated variable/s.

During the last years, shrinkage methods have become a commonly recognized and more effective methodology for solving the impact of the multicollinearity problem in several regression models. To solve this problem, Hoerl and Kennard [1, 2] proposed the ridge estimator. The concept of the ridge estimator is to add a small definite amount (k) to the diagonal entries of the covariance matrix to increase the conditioning of this matrix, reduce the mean squared error (MSE), and achieve consistent coefficients. For a review of the ridge estimator in both linear and GLMs, e.g., as shown in References Rady et al. [3], Abonazel and Taha [4], Qasim et al. [5], Alobaidi et al. [6], and Sami et al. [7].

One of the drawbacks of the ridge estimator is that estimated parameters are non-linear functions of the ridge parameter and the small k selected might not be high enough to solve multicollinearity. As a solution to this problem, Liu [8] developed the Liu estimator which is a linear function of the shrinkage parameter. The Liu estimator is a combination of the ridge estimator and the Stein estimator suggested by Stein [9]. For a review of the Liu estimator in both linear and GLMs, e.g., as shown in References. Liu [8], Karlsson et al. [10], Qasim et al. [11], and Naveed et al. [12]. Furthermore, Liu [13] proved the supremacy of the Liu-type estimator over the ridge and Liu estimators. Details about Liu-type estimator, properties, and applications in regression models are shown in References Liu [14], Özkale and Kaciranlar [15], Li and Yang [16], Kurnaz and Akay [17], Sahriman and Koerniawan [18], and Algamal and Abonazel [19]. As a good alternative for the Liu-type estimator, Özkale and Kaciranlar [15] proposed the two-parameter estimator, and they proved that the two-parameter estimator utilizes the power of both the ridge estimator and the Liu estimator. Extensions of two-parameter estimator in GLMs include Huang and Yang [20], Algamal [21], Asar and Genç [22], Rady et al. [23, 24], Çetinkaya and Kaçiranlar [25], Abonazel and Farghali [26], Akram et al. [27], and Lukman et al. [28].

The rest of the article is arranged as follows: Section Methodology presents an introduction about the beta regression model, its estimation using the ML method, and the proposed two-parameter estimator; Section Choosing the Shrinkage Parameters provides suggested shrinkage parameters for our estimator; Sections Simulation Study and Real Data Applications provide a numerical evaluation using both Monte Carlo simulation and two empirical data applications, respectively; and Section Conclusion offers some concluding remarks.

METHODOLOGY

Beta Regression Model

Practitioners usually use linear regression modeling to investigate the relationship and effect of some selected explanatory variables on the normal response variable. However, this is not suitable for circumstances where the response variable is constrained to the interval (0, 1) because it may give fitted values for the variable of concern that surpass its lower and upper limits. Therefore, inference based on the normality assumption can be deceptive. The beta regression model was first developed by Ferrari and Cribari-Neto [29] by connecting the mean function of its response variable to a set of linear predictors *via* a monotone differentiable function called the link function. This model contains a precision parameter, the inverse of which is called a dispersion scale. In the basic type of a beta regression model, the precision parameter is believed to be constant through observations. Nevertheless, the precision parameter might not be constant through findings such as those of Smithson and Verkuilen [30] and Cribari-Neto and Zeileis [31].

Let y is a continuous random variable that follows a beta distribution with the following probability density function:

$$\mathrm{f}\left(y;\ \mu,\phi\right) = \frac{\Gamma(\phi)}{\Gamma(\mu\phi)\,\Gamma((1-\mu)\,\phi)}\, y^{(\mu\phi)-1}\left(1-y\right)^{(\phi-\mu\phi-1)};\quad 0<y<1;\ 0<\mu<1;\ \phi>0, \tag{1}$$

where $\Gamma(\cdot)$ is the gamma function and ϕ is the precision parameter [32]:

$$\phi = \frac{1-\sigma^2}{\sigma^2}.$$

The mean and variance of the beta probability distribution are: $\mathrm{E}\left(y\right) = \mu$, $\mathrm{var}\left(y\right) = \mu\,(1-\mu)\,\sigma^2$. Using the logit link function, the model allows μ_i, depending on covariates as follows:

$$\mathrm{g}\,(\mu_i) = \log\left(\frac{\mu_i}{1-\mu_i}\right) = \mathrm{x}_i^T\beta = \eta_i, \tag{2}$$

where $\mathrm{g}(\cdot)$ be a monotonic differentiable link function used to relate the systematic component with the random component, $\beta = \left(\beta_1,\ldots,\beta_p\right)^T$ is a $p \times 1$ vector of unknown parameters, $\mathrm{x}_i = \left(x_{i1},\ldots,x_{ip}\right)^T$ is the vector of p regressors, and η_i is a linear predictor.

Estimation of the beta regression parameters is done by using the ML method [33]. The log-likelihood function of the beta regression model is given by:

$$\mathcal{L}\left(\mu_i,\phi;y_i\right) = \sum\nolimits_{i=1}^{n}\left\{\log\Gamma\,(\phi) - \log\Gamma\,(\mu_i\,(\phi)) - \log\Gamma\,((1-\mu_i)\,(\phi)) + (\mu_i\,(\phi)-1)\log\left(y_i\right) + ((1-\mu_i)\,(\phi)-1)\log\left(1-y_i\right)\right\} \tag{3}$$

By differentiating the log-likelihood function in Eq. (3) with respect to β , gives us the score function for β:

$$S\,(\beta) = \phi X^T A\left(y^* - \mu^*\right), \tag{4}$$

where $A = \mathrm{diag}\left(\frac{1}{\mathrm{g}'(\mu_1)},\ldots,\frac{1}{\mathrm{g}'(\mu_n)}\right)$, $y^* = \left(y_1^*,\ldots,y_n^*\right)^T$, $\mu^* = \left(\mu_1^*,\ldots,\mu_n^*\right)^T$, $y_i^* = \log\left(\frac{y_i}{1-y_i}\right)$, and $\mu_i^* = \psi\,(\mu_i\phi) - \psi\,((1-\mu_i)\,\phi)$, such that $\psi(\cdot)$ denoting the digamma function, and $g'(\cdot)$ is the first derivative of $g(\cdot)$. The iterative reweighted least-squares (IWLS) algorithm or Fisher's scoring algorithm was used for estimating β [34, 35]. The form of this algorithm can be written as:

$$\beta^{(r+1)} = \beta^{(r)} + \left(I_{\beta\beta}^{(r)}\right)^{-1} S_\beta^{(r)}\,(\beta),$$

where $S_{\beta}^{(r)}$ is the score function defined in Eq. (4), and $\mathrm{I}_{\beta\beta}^{(r)}$ is the information matrix for β , as shown in References Espinheira et al. [35] for more details. The initial value of β can be obtained by the least-squares estimation, while the initial value for each precision parameter is:

$$\hat{\phi}_i = \frac{\hat{\mu}_i\left(1-\hat{\mu}_i\right)}{\hat{\sigma}_i^2}, \tag{5}$$

where $\hat{\mu}$ and $\hat{\sigma}_i^2$ values are obtained from linear regression. Given $r = 0, 1, 2, \ldots$ is the number of iterations that are performed, convergence occurs when the difference between successive estimates becomes smaller than a given small constant. At the final step, the ML estimator of β is obtained as:

$$\hat{\beta}_{\mathrm{BML}} = \left(X^T\hat{W}X\right)^{-1}X^T\hat{W}\hat{z}, \tag{6}$$

where X is an $n\times p$ matrix of regressors, $\hat{z} = \hat{\eta}+\hat{W}^{-1}\hat{A}\left(y^{*}-\mu^{*}\right)$, and $\hat{W} = \mathrm{diag}\left(\hat{w}_1, \ldots, \hat{w}_n\right)$;

$$\hat{w}_i = \frac{\left(1-\hat{\sigma}^2\right)}{\hat{\sigma}^2}\left\{\psi'\left(\frac{\hat{\mu}_i\left(1-\hat{\sigma}^2\right)}{\hat{\sigma}^2}\right) + \psi'\left(\frac{\left(1-\hat{\mu}_i\right)\left(1-\hat{\sigma}^2\right)}{\hat{\sigma}^2}\right)\right\}\frac{1}{\left\{\mathrm{g}'\left(\hat{\mu}_i\right)\right\}^2},$$

where $\hat{W}$ and $\hat{A}$ are the estimated ML matrices of W and A, respectively. The ML estimator of β is normally distributed with asymptotic mean vectors $E\left(\hat{\beta}_{\mathrm{BR}}\right) = \beta$ and asymptotic covariance matrix:

$$\mathrm{Cov}\left(\hat{\beta}_{\mathrm{BML}}\right) = \frac{1}{\phi}\left(X^T\hat{W}X\right)^{-1} \tag{7}$$

Hence, the asymptotic trace mean squared error (TMSE) of $\hat{\beta}_{\mathrm{BML}}$ is

$$\mathrm{TMSE}\left(\hat{\beta}_{\mathrm{BML}}\right) = \mathrm{tr}\left[\frac{1}{\phi}(X^T\hat{W}X)^{-1}\right] \tag{8}$$

Ridge and Liu Estimators

Recently, Abonazel and Taha [4] and Qasim et al. [5] introduced the ridge beta regression (RBR) estimator as follows:

$$\hat{\beta}_{\mathrm{RBR}} = \left(X^T\hat{W}X+k\mathrm{I}\right)^{-1}X^T\hat{W}\hat{z};\ k>0 \tag{9}$$

It can note that if $k = 0$, then $\hat{\beta}_{\mathrm{RBR}} = \hat{\beta}_{\mathrm{BML}}$. The bias vector of the RBR estimator is

$$\mathrm{Bias}\left(\hat{\beta}_{\mathrm{RBR}}\right) = -k\left(X^T\hat{W}X+k\mathrm{I}\right)^{-1}\beta \tag{10}$$

Suppose that $\lambda_1 \geq \ldots \geq \lambda_p \geq 0$ are the ordered eigenvalues of $X^T\hat{W}X$ matrix and Q is the matrix whose columns are the eigenvectors of $X^T\hat{W}X$ matrix. Then $\Lambda = \mathrm{diag}\left(\lambda_1, \ldots, \lambda_p\right) = Q^TX^T\hat{W}XQ$ and $\alpha = Q^T\gamma$. Then, the matrix mean squared error (MMSE) of the RBR estimator is:

$$\begin{aligned}\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{RBR}}\right) &= \mathrm{Cov}\left(\hat{\beta}_{\mathrm{RBR}}\right)+\mathrm{Bias}\left(\hat{\beta}_{\mathrm{RBR}}\right)\mathrm{Bias}\left(\hat{\beta}_{\mathrm{RBR}}\right)^T\\ &= \frac{1}{\phi}\left(Q\Lambda_k^{-1}\Lambda\Lambda_k^{-1}Q^T\right)+k^2Q\Lambda_k^{-1}\alpha\alpha^T\Lambda_k^{-1}Q^T,\end{aligned} \tag{11}$$

where $\Lambda_k = \mathrm{diag}\left(\lambda_1+k, \ \ldots, \ \lambda_p+k\right)$, and the TMSE of the RBR estimator is

$$\begin{aligned}\mathrm{TMSE}\left(\hat{\beta}_{\mathrm{RBR}}\right) &= \mathrm{tr}\left(\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{RBR}}\right)\right)\\ &= \frac{1}{\phi}\sum\nolimits_{j=1}^{p}\frac{\lambda_j}{\left(\lambda_j+k\right)^2}+k^2\sum\nolimits_{j=1}^{p}\frac{\alpha_j^2}{\left(\lambda_j+k\right)^2}\end{aligned} \tag{12}$$

The first term in Eq. (12) is an asymptotic variance, and the second term is a square bias. Abonazel and Taha [4] and Qasim et al. [5] showed the derivation of the MSE properties of the RBR estimator.

The Liu estimator can be extended to the beta regression model, the Liu beta regression (LBR) estimator is given by Karlsson et al. [10] as:

$$\begin{aligned}\hat{\beta}_{\mathrm{LBR}} &= \left(X^T\hat{W}X+\mathrm{I}\right)^{-1}\left(X^T\hat{W}X+d\mathrm{I}\right)\hat{\beta}_{\mathrm{BML}};\\ &0<d<1,\end{aligned} \tag{13}$$

where d is the Liu parameter, the bias vector of the LBR estimator is:

$$\mathrm{Bias}\left(\hat{\beta}_{\mathrm{LBR}}\right) = \left(X^T\hat{W}X+\mathrm{I}\right)^{-1}\left(d-1\right)\beta \tag{14}$$

The MMSE for the LBR estimator can be derived as:

$$\begin{aligned}\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LBR}}\right) &= \mathrm{Cov}\left(\hat{\beta}_{\mathrm{LBR}}\right)+\mathrm{Bias}\left(\hat{\beta}_{\mathrm{LBR}}\right)\mathrm{Bias}\left(\hat{\beta}_{\mathrm{LBR}}\right)^T\\ &= \frac{1}{\phi}\left(Q\Lambda_1^{-1}\Lambda_d\Lambda^{-1}\Lambda_d\Lambda_1^{-1}Q^T\right)\\ &\quad+\left(d-1\right)^2Q\Lambda_1^{-1}\alpha\alpha^T\Lambda_1^{-1}Q^T,\end{aligned} \tag{15}$$

where $\Lambda_1 = \mathrm{diag}\left(\lambda_1+1, \ \ldots, \lambda_p+1\right)$ and $\Lambda_d = \mathrm{diag}\left(\lambda_1+d, \ \ldots, \ \lambda_p+d\right)$. The TMSE of the LBR estimator is:

$$\begin{aligned}\mathrm{TMSE}(\hat{\beta}_{\mathrm{LBR}}) &= \mathrm{tr}\left(\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LBR}}\right)\right)\\ &= \frac{1}{\phi}\sum_{j=1}^{p}\left\{\frac{\left(\lambda_j+d\right)^2}{\lambda_j\left(\lambda_j+1\right)^2}+\frac{\left(d-1\right)^2\alpha_j^2\phi}{\left(\lambda_j+1\right)^2}\right\}\end{aligned} \tag{16}$$

Recently, Algamal and Abonazel [19] developed the Liu-type beta regression (LTBR) estimator:

$$\begin{aligned}\hat{\beta}_{\mathrm{LTBR}} &= (X^T\hat{W}X+kI)^{-1}(X^T\hat{W}X-dI)\hat{\beta}_{\mathrm{BML}};\\ &k>0, -\infty<d<\infty\end{aligned} \tag{17}$$

The bias vector of the LTBR estimator is:

$$\text{Bias}\left(\hat{\beta}_{LTBR}\right) = -\left(d+k\right)\left(X^T\hat{W}X + k\text{I}\right)^{-1}\beta \tag{18}$$

The MMSE of the LTBR estimator is:

$$\begin{aligned}\text{MMSE}\left(\hat{\beta}_{\text{LTBR}}\right) &= \text{Cov}\left(\hat{\beta}_{\text{LTBR}}\right) + \text{Bias}\left(\hat{\beta}_{\text{LTBR}}\right)\text{Bias}\left(\hat{\beta}_{\text{LTBR}}\right)^T \\ &= \frac{1}{\phi}\left(Q\Lambda_k^{-1}\Lambda_{-d}\Lambda^{-1}\Lambda_{-d}\Lambda_k^{-1}Q^T\right) \\ &\quad +\left(d+k\right)^2 Q\Lambda_k^{-1}\alpha\alpha^T\Lambda_k^{-1}Q^T \end{aligned}\tag{19}$$

where $\Lambda_{-d} = \text{diag}\left(\lambda_1 - d, \ \ldots, \ \lambda_p - d\right)$. Then, the MSE of the LTBR estimator is

$$\begin{aligned}\text{MSE}\left(\hat{\beta}_{\text{LTBR}}\right) &= \text{tr}\left(\text{MMSE}\left(\hat{\beta}_{\text{LTBR}}\right)\right) \\ &= \frac{1}{\phi}\sum_{j=1}^{p}\left\{\frac{\left(\lambda_j - d\right)^2}{\lambda_j\left(\lambda_j + k\right)^2} + \frac{\left(d+k\right)^2\alpha_j^2\,\phi}{\left(\lambda_j + k\right)^2}\right\}\end{aligned}\tag{20}$$

The Proposed Estimator

In this section, we extend the two-parameter estimator introduced by Özkale and Kaçiranlar [15] to the beta regression model to combat multicollinearity and obtain more stable and accurate results. The two-parameter beta regression (TPBR) estimator can be written as follows:

$$\begin{aligned}\hat{\beta}_{\text{TPBR}} &= (X^T\hat{W}X + kI)^{-1}(X^T\hat{W}X + kdI)\hat{\beta}_{\text{BML}}; \\ &\quad k > 0; 0 < d < 1\end{aligned}\tag{21}$$

It is worth noting that the TPBR estimator is a general class that has some estimators as special cases. These estimators are the LBR, RBR, and beta maximum likelihood (BML) estimators, which can be given, respectively, as follows:

$$\begin{aligned}\lim_{k\to 1}\hat{\beta}_{\text{TPBR}} &= \hat{\beta}_{\text{LBR}} = \left(X^T\hat{W}X + I\right)^{-1}\left(X^T\hat{W}X + dI\right)\hat{\beta}_{\text{BML}}, \\ \lim_{d\to 0}\hat{\beta}_{\text{TPBR}} &= \hat{\beta}_{RBR} = \left(X^T\hat{W}X + kI\right)^{-1}\left(X^T\hat{W}X\right)\hat{\beta}_{\text{BML}}, \\ \lim_{k\to 0}\hat{\beta}_{\text{TPBR}} &= \hat{\beta}_{\text{BML}} = \left(X^T\hat{W}X\right)^{-1}(X^T\hat{W}\hat{z}).\end{aligned}$$

The bias vector of the TPBR estimator is

$$\text{Bias}\left(\hat{\beta}_{\text{TPBR}}\right) = k\left(d-1\right)\left(X^T\hat{W}X + k\text{I}\right)^{-1}\beta \tag{22}$$

The MMSE for TPBR estimator can be derived as:

$$\begin{aligned}\text{MMSE}\left(\hat{\beta}_{\text{TPBR}}\right) &= \text{Cov}\left(\hat{\beta}_{\text{TPBR}}\right) + \text{Bias}\left(\hat{\beta}_{\text{TPBR}}\right)\text{Bias}\left(\hat{\beta}_{\text{TPBR}}\right)^T \\ &= \frac{1}{\phi}\left(Q\Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}Q^T\right) \\ &\quad +k^2\left(d-1\right)^2 Q\Lambda_k^{-1}\alpha\alpha^T\Lambda_k^{-1}Q^T,\end{aligned}\tag{23}$$

where $\Lambda_{kd} = \text{diag}\left(\lambda_1 + kd, \ \lambda_2 + kd, \ \ldots, \lambda_p + kd\right)$, the TMSE of the TPBR estimator is:

$$\begin{aligned}\text{TMSE}\left(\hat{\beta}_{\text{TPBR}}\right) &= \text{tr}\left(\text{MMSE}\left(\hat{\beta}_{\text{TPBR}}\right)\right) \\ &= \frac{1}{\phi}\sum_{j=1}^{p}\left\{\frac{\left(\lambda_j + kd\right)^2}{\lambda_j\left(\lambda_j + k\right)^2} + \frac{\alpha_j^2\phi k^2\left(d-1\right)^2}{\left(\lambda_j + k\right)^2}\right\}\end{aligned}\tag{24}$$

The Superiority of the New Estimator

The following lemmas prove the superiority of the two-parameter beta estimator over the other estimators.

Lemma 1. Farebrother [36]: Let M be a positive definite matrix, δ be a vector of non-zero constants, and c be a positive constant. Then, $cM - \delta\delta^T > 0$ if and only if (iff) $\delta M\delta^T < c$.

Two-Parameter Beta Estimator vs. ML Estimator

The following lemma gives the condition that the TPBR estimator is superior to the ML estimator:

Lemma 2. under the beta regression model, let $k > 0$, $0 < d < 1$, and $b_{\text{TPBR}} = \text{Bias}\left(\hat{\beta}_{\text{TPBR}}\right)$. Then, $\text{MMSE}\left(\hat{\beta}_{\text{BML}}\right) - \text{MMSE}\left(\hat{\beta}_{\text{TPBR}}\right) > 0$ iff $k\left(1-d\right)\left(2\lambda_j + k\left(1+d\right)\right) > 0$.

Proof: the difference between the MMSE functions of the ML estimator and the TPBR estimator is obtained by:

$$\begin{aligned}&\text{MMSE}\left(\hat{\beta}_{\text{BML}}\right) - \text{MMSE}\left(\hat{\beta}_{\text{TPBR}}\right) \\ &= \frac{1}{\phi}\,Q\left(\phi\Lambda^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}\right)Q^T - b_{\text{TPBR}}b_{\text{TPBR}}^T,\end{aligned}$$

The matrix $\left(\phi\Lambda^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}\right)$ is positive definite, if $\phi\left(\lambda_j + k\right)^2 - \left(\lambda_j + kd\right)^2 > 0$, which is equivalent to $\left[\left(\lambda_j + k\right) + \left(\lambda_j + kd\right)\right]\left[\left(\lambda_j + k\right) - \left(\lambda_j + kd\right)\right] > 0$. Simplifying the last inequality, one gets $k\left(1-d\right)\left(2\lambda_j + k\left(1+d\right)\right) > 0$. The proof is finished by Lemma 1.

Two-Parameter Estimator vs. Ridge Estimator

The following lemma gives that the TPBR estimator is superior to the RBR estimator:

Lemma 3. under the beta regression model, consider $k > 0$, $0 < d < 1$, and $b_{\text{RBR}} = \text{Bias}\left(\hat{\beta}_{\text{RBR}}\right)$. Then, $\text{MMSE}\left(\hat{\beta}_{\text{RBR}}\right) - \text{MMSE}\left(\hat{\beta}_{\text{TPBR}}\right) > 0$ iff $kd\left(2\lambda_j + kd\right) > 0$.

Proof: the difference between the MMSE functions of the RBR estimator and the TPBR estimator is obtained by:

$$\begin{aligned}&\text{MMSE}\left(\hat{\beta}_{\text{RBR}}\right) - \text{MMSE}\left(\hat{\beta}_{\text{TPBR}}\right) \\ &= \frac{1}{\phi}Q\left(\Lambda_k^{-1}\Lambda\Lambda_k^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}\right) \\ &\quad Q^T + b_{\text{RBR}}b_{\text{RBR}}^T - b_{\text{TPBR}}b_{\text{TPBR}}^T,\end{aligned}$$

This can be rewritten as:

$$\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{RBR}}\right) - \mathrm{MMSE}\left(\hat{\beta}_{\mathrm{TPBR}}\right) = \frac{1}{\phi} Q \,\mathrm{diag}\left\{\frac{\lambda_j}{\left(\lambda_j + k\right)^2} - \frac{\left(\lambda_j + kd\right)^2}{\lambda_j\left(\lambda_j + k\right)^2}\right\}_{j=1}^{p} Q^T + b_{\mathrm{RBR}} b_{\mathrm{RBR}}^T - b_{\mathrm{TPBR}} b_{\mathrm{TPBR}}^T,$$

The matrix $\Lambda_k^{-1}\Lambda\Lambda_k^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}$ is positive definite if $\lambda_j^2 - \left(\lambda_j + kd\right)^2 > 0$ which is equivalent to $\left[\lambda_j - \left(\lambda_j + kd\right)\right]\left[\lambda_j + \left(\lambda_j + kd\right)\right] > 0$. Simplifying the last inequality, one gets $kd\left(2\lambda_j + kd\right) > 0$. Then, using Lemma 1, the proof is finished.

Two-Parameter Estimator vs. Liu Estimator

The following lemma gives the condition that the TPBR estimator is superior to the LBR estimator:

Lemma 4. under the beta regression model, consider $k > 0$, $0 < d < 1$, and $b_{\mathrm{LBR}} = \mathrm{Bias}\left(\hat{\beta}_{\mathrm{LBR}}\right)$. Then, $\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LBR}}\right) - \mathrm{MMSE}\left(\hat{\beta}_{\mathrm{TPBR}}\right) > 0$ iff $\left(\lambda_j + d\right)^2\left(\lambda_j + k\right)^2 - \left(\lambda_j + kd\right)^2\left(\lambda_j + I\right)^2 > 0$.

Proof: the difference between the MMSE functions of $\hat{\beta}_{\mathrm{LBR}}$ and $\hat{\beta}_{\mathrm{TPBR}}$ is obtained by:

$$\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LBR}}\right) - \mathrm{MMSE}\left(\hat{\beta}_{\mathrm{TPBR}}\right) = \frac{1}{\phi} Q\left(\Lambda_1^{-1}\Lambda_d\Lambda^{-1}\Lambda_d\Lambda_1^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}\right) Q^T + b_{\mathrm{LBR}} b_{\mathrm{LBR}}^T - b_{\mathrm{TPBR}} b_{\mathrm{TPBR}}^T,$$

This can be rewritten as:

$$\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LBR}}\right) - \mathrm{MMSE}\left(\hat{\beta}_{\mathrm{TPBR}}\right) = \frac{1}{\phi} Q \,\mathrm{diag}\left\{\frac{\left(\lambda_j + d\right)^2}{\lambda_j\left(\lambda_j + I\right)^2} - \frac{\left(\lambda_j + kd\right)^2}{\lambda_j\left(\lambda_j + k\right)^2}\right\}_{j=1}^{p} Q^T + b_{\mathrm{LBR}} b_{\mathrm{LBR}}^T - b_{\mathrm{TPBR}} b_{\mathrm{TPBR}}^T,$$

The matrix $\Lambda_1^{-1}\Lambda_d\Lambda^{-1}\Lambda_d\Lambda_1^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}$ is positive definite if $\left(\lambda_j + d\right)^2\left(\lambda_j + k\right)^2 - \left(\lambda_j + kd\right)^2\left(\lambda_j + I\right)^2 > 0$, which is equivalent to $\left[\left(\lambda_j + d\right)^2\left(\lambda_j + k\right)^2 > \left(\lambda_j + kd\right)^2\left(\lambda_j + I\right)^2\right]$. For $k > 0, 0 < d < 1$, it can be observed that $\left(\lambda_j + d\right)^2\left(\lambda_j + k\right)^2 - \left(\lambda_j + kd\right)^2\left(\lambda_j + I\right)^2 > 0$. The proof is finished by Lemma 1.

Two-Parameter Estimator vs. Liu-Type Estimator

The following lemma gives the condition that the TPBR estimator is superior to the LTBR estimator:

Lemma 5. under the beta regression model, consider $k > 0$, $-\infty < d_1 < \infty$, $0 < d_2 < 1$, and $b_{\mathrm{LTBR}} = \mathrm{Bias}\left(\hat{\beta}_{\mathrm{LTBR}}\right)$, where d_1 and d_2 are the d values of LTBR and TPBR estimators, respectively. Then, $\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LTBR}}\right) - \mathrm{MMSE}\left(\hat{\beta}_{\mathrm{TPBR}}\right) > 0$ iff $d_1\left(d_1 - 2\lambda_j\right) - kd_2\left(kd_2 + 2\lambda_j\right) > 0$.

Proof: the difference between the MMSE functions of $\hat{\beta}_{\mathrm{LTBR}}$ and $\hat{\beta}_{\mathrm{TPBR}}$ is obtained by:

$$\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LTBR}}\right) - \mathrm{MMSE}\left(\hat{\beta}_{\mathrm{TPBR}}\right) = \frac{1}{\phi} Q\left(\Lambda_k^{-1}\Lambda_{-d}\Lambda^{-1}\Lambda_{-d}\Lambda_k^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}\right) Q^T + b_{\mathrm{LTBR}} b_{\mathrm{LTBR}}^T - b_{\mathrm{TPBR}} b_{\mathrm{TPBR}}^T.$$

This can be rewritten as:

$$\mathrm{MMSE}\left(\hat{\beta}_{\mathrm{LTBR}}\right) - \mathrm{MMSE}\left(\hat{\beta}_{\mathrm{TPBR}}\right) = \frac{1}{\phi} Q \,\mathrm{diag}\left\{\frac{\left(\lambda_j - d_1\right)^2}{\lambda_j\left(\lambda_j + k\right)^2} - \frac{\left(\lambda_j + kd_2\right)^2}{\lambda_j\left(\lambda_j + k\right)^2}\right\}_{j=1}^{p} Q^T + b_{\mathrm{LTBR}} b_{\mathrm{LTBR}}^T - b_{\mathrm{TPBR}} b_{\mathrm{TPBR}}^T,$$

The matrix $\Lambda_k^{-1}\Lambda_{-d}\Lambda^{-1}\Lambda_{-d}\Lambda_k^{-1} - \Lambda_k^{-1}\Lambda_{kd}\Lambda^{-1}\Lambda_{kd}\Lambda_k^{-1}$ is positive definite if $\left(\lambda_j - d_1\right)^2 - \left(\lambda_j + kd_2\right)^2 > 0$, which is equivalent to $\left[\left(\lambda_j - d_1\right)^2 > \left(\lambda_j + kd_2\right)^2\right]$. For $k > 0, -\infty < d_1 < \infty, 0 < d_2 < 1$, it can be observed that $d_1\left(d_1 - 2\lambda_j\right) - kd_2\left(kd_2 + 2\lambda_j\right) > 0$. The proof is finished by Lemma 1.

CHOOSING THE SHRINKAGE PARAMETERS

There is no definite rule for estimating the shrinkage parameters (k and d). However, we propose some methods based on the work of Hoerl et al. [37] and Kibria [38]. For the RBR estimator, we can use the k parameter of Hoerl and Kennard [1] after modifying their formula based on the optimal k of the beta regression model [5]:

$$k = \frac{1}{\phi \sum_{j=1}^{p} \hat{\alpha}_j^2}, \tag{25}$$

where $\hat{\alpha}_j$ is the jth element of the vector $\hat{\alpha} = Q^T \hat{\beta}_{\mathrm{BML}}$.

For the LBR estimator, we can use the optimal d parameter proposed by Karlsson et al. [10]:

$$d = \frac{\sum_{j=1}^{p}\left[\left(\hat{\alpha}_j^2 - \frac{1}{\phi}\right)/\left(\lambda_j + 1\right)^2\right]}{\sum_{j=1}^{p}\left[\left(\frac{1}{\phi\lambda_j} + \hat{\alpha}_j^2\right)/\left(\lambda_j + 1\right)^2\right]} \tag{26}$$

For the LTBR estimator, we can use the optimal d parameter of the LTBR estimator that was proposed by Algamal and Abonazel [19]:

$$d_{LTBR} = \frac{\sum_{j=1}^{p}\left[\left(\frac{1}{\phi} - k\hat{\alpha}_j^2\right)/\left(\lambda_j + k\right)^2\right]}{\sum_{j=1}^{p}\left[\left(\frac{1}{\phi} + \lambda_j\hat{\alpha}_j^2\right)/\lambda_j\left(\lambda_j + k\right)^2\right]} \tag{27}$$

Since d_{LTBR} depends on k, we suggest using the k parameter in Eq. (25).

For the proposed estimator (TPBR), we start by taking the derivative of MSE function given in Eq. (24) with respect to k and equating the resulting function to zero and by solving for the parameter k, we obtain the following individual parameters:

$$k_j = \frac{\lambda_j}{\phi\left(\lambda_j\hat{\alpha}_j^2\left(1-d\right)-\left(d/\phi\right)\right)}; j = 1,\ldots,p. \tag{28}$$

Since each individual parameter k_j should be positive, we obtain the following upper bound for the k_j parameter d's so that $k_j > 0$:

$$d < \ min\left(\frac{\lambda_j\hat{\alpha}_j^2}{\frac{1}{\phi}+\lambda_j\hat{\alpha}_j^2}\right)_{j=1}^{p}, \tag{29}$$

where $min(\cdot)$ is the minimum function such that $0 < d < 1$ and $\hat{\alpha}_j$ is the jth element of the vector $\hat{\alpha}$. Therefore, we propose the following shrinkage parameters for the TPBR estimator:

$$d_{\text{TPBR}} = \frac{1}{2}\ min\left(\frac{\lambda_j\hat{\alpha}_j^2}{\frac{1}{\phi}+\lambda_j\hat{\alpha}_j^2}\right)_{j=1}^{p}; \tag{30}$$

$$k_{\text{TPBR}} = \frac{1}{p}\sum\nolimits_{j=1}^{p}\left(\frac{\lambda_j}{\phi\left(\lambda_j\hat{\alpha}_j^2\left(1-d_{\text{TPBR}}\right)-d_{\text{TPBR}/\phi}\right)}\right) \tag{31}$$

Note that d_{TPBR} in Eq. (30) is always <1 and bigger than zero, and k_{TPBR} in Eq. (31) is always positive [15].

SIMULATION STUDY

A Monte Carlo simulation study has been conducted to compare the performances of BML, RBR, LBR, and LTBR estimators with the proposed estimator (TPBR estimator). Our simulation study is computed based on R-software, using the "*betareg*" package.

Simulation Design

The response variable y_i is generated as $y_i \sim \ Beta(\mu_i, \phi)$, with $\phi \ \in \{0.5,1,1.5\}$ and $\mu_i \ = \ \exp(x_i^T\beta)/(1 + \exp(x_i^T\beta))$ for $i \ =$

TABLE 1 | Mean squared error (MSE) values for different estimators when n = 50.

ϕ	p	ρ	BML	RBR	LBR	LTBR	TPBR
0.5	4	0.90	2.259	1.906	1.828	1.8026	1.419
		0.95	3.356	2.391	2.117	2.0916	2.033
		0.99	4.399	2.908	2.175	2.1496	2.134
	8	0.90	2.376	1.736	1.487	1.4616	1.312
		0.95	4.421	2.567	2.011	1.9856	1.862
		0.99	5.353	3.292	2.518	2.4926	1.669
	12	0.90	3.613	1.364	1.189	1.1636	1.013
		0.95	6.836	2.715	1.987	1.9616	1.107
		0.99	9.375	3.042	1.814	1.7886	1.078
1	4	0.90	1.951	1.598	1.520	1.4946	1.111
		0.95	3.048	2.083	1.809	1.7836	1.725
		0.99	4.091	2.601	1.867	1.8416	1.826
	8	0.90	2.068	1.428	1.179	1.1536	1.004
		0.95	4.112	2.259	1.703	1.6776	1.554
		0.99	5.045	2.984	2.210	2.1846	1.361
	12	0.90	3.305	1.056	0.881	0.8556	0.705
		0.95	6.528	2.407	1.679	1.6536	0.799
		0.99	9.067	2.734	1.506	1.4806	0.771
1.5	4	0.90	1.829	1.476	1.398	1.3726	0.989
		0.95	2.926	1.961	1.687	1.6616	1.603
		0.99	3.969	2.478	1.745	1.7196	1.704
	8	0.90	1.946	1.306	1.057	1.0316	0.882
		0.95	3.990	2.137	1.581	1.5556	1.432
		0.99	4.923	2.862	2.088	2.0626	1.239
	12	0.90	3.183	0.934	0.759	0.7336	0.583
		0.95	6.406	2.285	1.557	1.5316	0.677
		0.99	8.945	2.612	1.384	1.3586	0.648

TABLE 2 | Mean squared error values for different estimators when n = 100.

ϕ	p	ρ	BML	RBR	LBR	LTBR	TPBR
0.5	4	0.90	2.212	1.859	1.782	1.7566	1.372
		0.95	3.309	2.343	2.071	2.0456	1.985
		0.99	4.352	2.861	2.128	2.1026	2.087
	8	0.90	2.328	1.689	1.439	1.4136	1.265
		0.95	4.372	2.519	1.964	1.9386	1.814
		0.99	5.306	3.244	2.471	2.4456	1.621
	12	0.90	3.565	1.317	1.142	1.1166	0.965
		0.95	6.788	2.668	1.941	1.9156	1.063
		0.99	9.327	2.994	1.766	1.7406	1.031
1	4	0.90	1.904	1.551	1.472	1.4466	1.064
		0.95	3.001	2.035	1.762	1.7366	1.677
		0.99	4.044	2.553	1.822	1.7966	1.779
	8	0.90	2.022	1.381	1.131	1.1056	0.957
		0.95	4.064	2.211	1.656	1.6306	1.506
		0.99	4.998	2.936	2.162	2.1366	1.313
	12	0.90	3.257	1.009	0.834	0.8086	0.657
		0.95	6.483	2.364	1.632	1.6066	0.752
		0.99	9.019	2.686	1.458	1.4326	0.723
1.5	4	0.90	1.782	1.429	1.355	1.3296	0.942
		0.95	2.879	1.913	1.647	1.6216	1.555
		0.99	3.922	2.431	1.698	1.6726	1.657
	8	0.90	1.898	1.259	1.009	0.9836	0.835
		0.95	3.942	2.089	1.534	1.5086	1.384
		0.99	4.876	2.814	2.041	2.0156	1.191
	12	0.90	3.135	0.887	0.712	0.6866	0.535
		0.95	6.358	2.238	1.513	1.4876	0.631
		0.99	8.897	2.564	1.336	1.3106	0.601

$1, 2, \ldots, n$, and $\beta = \left(\beta_1, \ldots, \beta_p\right)^T$ with $\sum_{j=1}^{p} \beta_j^2 = 1$ and $\beta_1 = \ldots = \beta_p$ [19, 26, 39–41].

The explanatory variables $x_i = \left(x_{i1}, \ldots, x_{ip}\right)^T$ are generated from the following:

$$x_{ij} = (1 - \rho^2)^{0.5} w_{ij} + \rho w_{ip},\ i = 1, 2, \ldots, n, \quad j = 1, 2, \ldots, p, \tag{32}$$

where ρ is the coefficient of the correlation between the explanatory variables and w_{ij} are independent standard normal pseudo-random numbers.

It is well-known that the sample size (n), the number of explanatory variables (p), and the pairwise correlation (ρ) between the explanatory variables have a direct impact on the prediction accuracy. Therefore, four values of n are considered: 50, 100, 250, and 400. In addition, three values of p are considered: 4, 8, and 12. Further, three values of ρ are considered: 0.90, 0.95, and 0.99. For a combination of these different values of n, ϕ, p, and ρ, the generated data are repeated $L = 1,000$ times and the average MSE is calculated as:

$$MSE\left(\hat{\beta}\right) = \frac{1}{L}\sum_{l=1}^{L} \left(\hat{\beta}_l - \beta\right)^T \left(\hat{\beta}_l - \beta\right), \tag{33}$$

where $\hat{\beta}_l$ is the estimated vector of β.

Simulation Results

The averaged MSE for all the combinations of $n, \phi, p,$, and ρ are summarized in **Tables 1–4**. According to the simulation results, we conclude the following:

1. The TPBR estimator has the best performance in all the situations considered. Moreover, the performance of the TPBR estimator is better for larger values of ρ.
2. It is noted from **Tables 1–4** that the TPBR estimator ranks first with respect to MSE. In the second rank is the LTBR estimator, as it performs better than BML, RBR, and LBR estimators. Additionally, the BML estimator has the worst performance among RBR, LBR, and TPBR estimators which is significantly impacted by the multicollinearity.
3. Regarding the number of explanatory variables, it is easily seen that there is a negative impact on MSE, where there are increases in their values when the p increase from four variables to eight and twelve variables. In addition, in terms of the sample size, the MSE values decrease when n increases, regardless of the value of ρ, ϕ, and p.
4. Clearly, the MSE values are decreasing when ϕ is increasing.

TABLE 3 | Mean squared error values for different estimators when $n = 250$.

ϕ	p	ρ	BML	RBR	LBR	LTBR	TPBR
0.5	4	0.90	2.151	1.798	1.719	1.6936	1.311
		0.95	3.248	2.282	2.009	1.9836	1.924
		0.99	4.291	2.801	2.067	2.0416	2.026
	8	0.90	2.267	1.628	1.378	1.3526	1.204
		0.95	4.311	2.458	1.903	1.8776	1.753
		0.99	5.245	3.183	2.409	2.3836	1.561
	12	0.90	3.504	1.256	1.081	1.0556	0.904
		0.95	6.727	2.607	1.879	1.8536	0.999
		0.99	9.266	2.933	1.705	1.6796	0.972
1	4	0.90	1.843	1.491	1.411	1.3856	1.003
		0.95	2.942	1.974	1.701	1.6756	1.616
		0.99	3.983	2.492	1.759	1.7336	1.718
	8	0.90	1.959	1.325	1.073	1.0476	0.896
		0.95	4.003	2.154	1.595	1.5696	1.445
		0.99	4.937	2.875	2.101	2.0756	1.252
	12	0.90	3.196	0.948	0.773	0.7476	0.596
		0.95	6.419	2.299	1.571	1.5456	0.691
		0.99	8.958	2.625	1.397	1.3716	0.662
1.5	4	0.90	1.721	1.368	1.289	1.2636	0.881
		0.95	2.818	1.852	1.579	1.5536	1.494
		0.99	3.861	2.372	1.637	1.6116	1.596
	8	0.90	1.837	1.198	0.948	0.9226	0.774
		0.95	3.881	2.028	1.473	1.4476	1.323
		0.99	4.815	2.753	1.979	1.9536	1.131
	12	0.90	3.074	0.826	0.651	0.6256	0.474
		0.95	6.297	2.177	1.449	1.4236	0.569
		0.99	8.836	2.503	1.275	1.2496	0.547

TABLE 4 | Mean squared error values for different estimators when $n = 400$.

ϕ	p	ρ	BML	RBR	LBR	LTBR	TPBR
0.5	4	0.90	2.117	1.764	1.685	1.2596	1.277
		0.95	3.214	2.248	1.975	1.7496	1.891
		0.99	4.257	2.767	2.033	2.0076	1.992
	8	0.90	2.233	1.594	1.344	1.3186	1.172
		0.95	4.277	2.424	1.869	1.3436	1.719
		0.99	5.211	3.149	2.375	2.3496	1.527
	12	0.90	3.47	1.222	1.047	1.0216	0.871
		0.95	6.693	2.573	1.845	1.8196	0.965
		0.99	9.232	2.899	1.671	1.6456	0.938
1	4	0.90	1.809	1.457	1.377	1.3516	0.969
		0.95	2.908	1.94	1.667	1.6416	1.582
		0.99	3.949	2.458	1.725	1.6996	1.684
	8	0.90	1.925	1.291	1.039	1.0136	0.862
		0.95	3.969	2.12	1.561	1.5356	1.411
		0.99	4.903	2.841	2.067	2.0416	1.218
	12	0.90	3.162	0.914	0.739	0.7136	0.562
		0.95	6.385	2.265	1.537	1.5116	0.657
		0.99	8.924	2.591	1.363	1.3376	0.628
1.5	4	0.90	1.687	1.334	1.255	1.2296	0.847
		0.95	2.784	1.818	1.545	1.5196	1.462
		0.99	3.827	2.338	1.603	1.5776	1.562
	8	0.90	1.803	1.164	0.914	0.8886	0.742
		0.95	3.847	1.994	1.439	1.4136	1.289
		0.99	4.781	2.719	1.945	1.9196	1.097
	12	0.90	3.04	0.792	0.617	0.5916	0.448
		0.95	6.263	2.143	1.415	1.3896	0.535
		0.99	8.802	2.469	1.241	1.2156	0.513

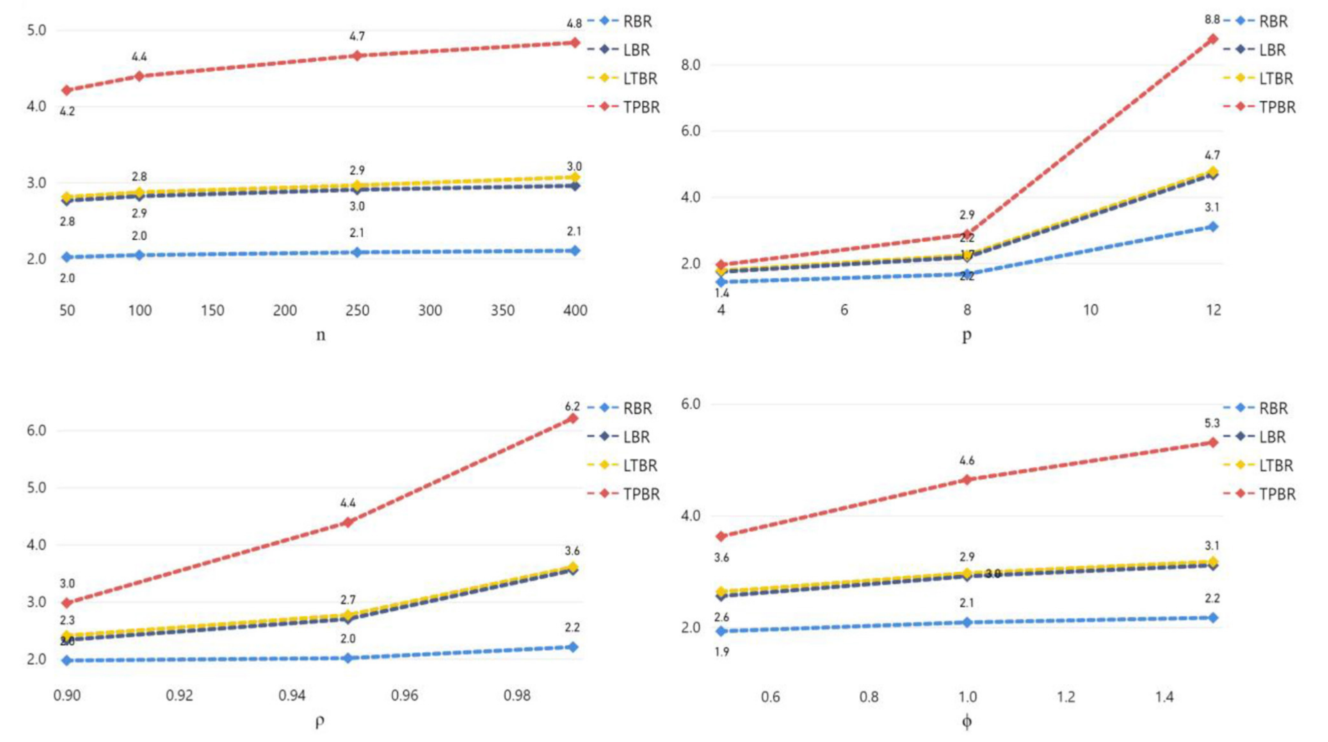


FIGURE 1 | Relative efficiency (RE) of different estimators categorized by levels of n, p, ρ, and ϕ.

TABLE 5 | The estimated coefficients and MSE values for the used estimators (football data).

	BML	RBR	LBR	LTBR	TPBR
x_1	−0.01749	−0.01761	−0.026786	−0.002165	−0.003165
x_2	0.026057	0.026399	0.052970	0.000354	0.000254
x_3	0.030190	0.030276	0.036945	0.000701	−0.000807
x_4	−0.032857	−0.031889	0.043208	−0.000355	0.000323
x_5	−0.129230	−0.128710	−0.088372	−0.002704	−0.002633
x_6	1.643973	1.629061	0.472132	0.021849	0.028245
MSE	0.04345	0.018314	0.006583	0.005208	0.005085

Relative Efficiency

Another comparative performance called relative efficiency (RE) can be utilized, it is calculated based on the MSE in Eq. (33) as follows [4, 39]:

$$\mathrm{RE}\left(\hat{\beta}_S\right) = \frac{MSE\left(\hat{\beta}_{BML}\right)}{MSE\left(\hat{\beta}_S\right)}, \tag{34}$$

where $\hat{\beta}_S$ denotes the estimators of RBR, LBR, LTBR, or TPBR. The RE results are shown in **Figure 1**.

Figure 1 shows that the RE of the four biased (RBR, LBR, LTBR, and TPBR) estimators were increased if the sample size (n), the number of explanatory variables (p), the degree of correlation between explanatory variables (ρ), and/or the precision parameter value (ϕ) are increased. Moreover, we can observe that the TPBR estimator has higher RE values than the other estimators.

TABLE 6 | The estimated coefficients and MSE values for the used estimators (gasoline yield data).

	BML	RBR	LBR	LTBR	TPBR
Gravity	−0.01749	−0.01761	−0.026786	−0.00571	−0.003165
Pressure	0.026057	0.026399	0.052970	0.04212	0.000254
Temp10	0.030190	0.030276	0.036945	0.04284	−0.000807
Temp	−0.032857	−0.031889	0.043208	0.00235	0.000323
MSE	0.04345	0.018314	0.006583	0.00537	0.005085

REAL DATA APPLICATIONS

In this section, we used two real data applications to investigate the advantage of our proposed (TPBR) estimator in different fields.

Football Spanish Data

We apply the proposed estimator to the football Spanish La Liga, season 2016–2017 [19]. The data contain 20 teams. The response variable is the proportion of won matches. The six considerable explanatory variables are: x_1 is the number of yellow cards, x_2 is the number of red cards, x_3 is the total number of substitutions, x_4 is the number of matches with 2.5 goals on average, x_5 is the number of matches that ended with goals, and x_6 is the ratio of the goal scores to the number of matches.

First, to check whether there is a multicollinearity problem or not, the correlation matrix and condition number (CN) are used. Based on the correlation matrix among the six explanatory variables that are presented, displayed by Algamal and Abonazel [19]. It is obviously seen that there are correlations >0.82 between x_1 and x_6, x_1 and x_4, x_2 and x_4, and x_4 and x_6. Second, the condition number, $\mathrm{CN} = \sqrt{\lambda_{\max}/\lambda_{\min}}$ of the data is 806.63 indicating the existence of multicollinearity. The estimated beta regression coefficients and MSE values for the BML, RBR, LBR, LTBR, and TPBR estimators are recorded in **Table 5**. From **Table 5**, it can note that the estimated coefficients of all estimators have the same signs; this means that the type of relationship between each explanatory variable and the response variable is not changed from what it was in the BML estimator. But MSE values of RBR, LBR, LTBR, and TPBR estimators are lower than the BML estimator. Whereas, the MSE value of the TPBR estimator is the lowest.

Gasoline Yield Data

To further investigate the advantage of our proposed estimator (TPBR), we apply the TPBR estimator to the chemical dataset (gasoline yield data) which was originally obtained by Prater [42], and later used by the following authors: Ospina et al. [43] and Karlsson et al. [10]. The dataset contains 32 observations on the response and four explanatory variables. The variables in the study are described as follows: the dependent variable y is the proportion of crude oil after distillation and fractionation while the explanatory variables are crude oil gravity (Gravity), vapor pressure of crude oil (Pressure), temperature at which 10% of the crude oil has vaporized (Temp10), and temperature at which all petrol in the amount of crude oil vaporizes (Temp). Atkinson [44] analyzed this dataset using the linear regression model and observed some anomalies in the distribution of the error. Recently, Karlsson et al. [10] showed that the beta regression model is more suitable to model the data.

The CN for the dataset under study is 11,281.4, which signals severe multicollinearity. The estimated beta regression coefficients and MSE values for the used estimators are recorded in **Table 6**. From **Table 6**, it can be noted that the estimated coefficients of all estimators have the same signs. In addition, MSE values of RBR, LBR, LTBR, and TPBR estimators are lower than the BML estimator. Whereas, the MSE value of the TPBR estimator is the lowest.

CONCLUSION

This article provided a two-parameter (TPBR) estimator for the beta regression model as a remedy for a multicollinearity problem. We proved, theoretically, that our proposed estimator is efficient than other biased estimators (ridge, Liu, and Liu-type estimators) suggested in the literature. Furthermore, a Monte Carlo simulation study was conducted to study the performance of the proposed estimator and ML, ridge, Liu, and Liu-type estimators. The results indicated that the proposed estimator outperforms these estimators, especially when there is a high-to-strong correlation between the explanatory variables. Finally, the benefit is shown by two empirical applications where the TPBR estimator performed well by decreasing the MSE compared with the ML, ridge, Liu, and Liu-type estimators.

For future work, for example, one can study the high-dimensional case in beta regression as an extension to Arashi et al. [45] or provide a robust biased estimation of beta regression as an extension to Awwad et al. [41] and Dawoud and Abonazel [40].

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found at: For first application: https://www.laliga.es. For second application: the R-package betareg.

AUTHOR CONTRIBUTIONS

MA: methodology, relative efficiency, interpreting the results, abstract, conclusions, writing—original draft, and final revision. ZA: simulation study and applications, interpreting the results, and revision. FA: methodology, interpreting the results, and revision. IT: introduction, methodology, and writing—original draft. All authors contributed to the article and approved the submitted version.

ACKNOWLEDGMENTS

The authors would like to thank the Deanship of Scientific Research at King Saud University represented by the Research Center at CBA for supporting this research financially.

REFERENCES

1. Hoerl AE, Kennard RW. Ridge regression: biased estimation for non-orthogonal problems. *Technometrics.* (1970) 12:55–67. doi: 10.1080/00401706.1970.10488634
2. Hoerl AE, Kennard RW. Ridge regression: applications to non-orthogonal problems. *Technometrics.* (1970) 12:69–82. doi: 10.1080/00401706.1970.10488635
3. Rady EA, Abonazel MR, Taha IM. Ridge estimators for the negative binomial regression model with application. In: *The 53rd Annual Conference on Statistics, Computer Science, and Operation Research 3-5 Dec, 2018.* (2018). Giza: ISSR, Cairo University.
4. Abonazel MR, Taha IM. Beta ridge regression estimators: simulation and application. *Commun Statist Simul Computat.* (2021) 1–13. doi: 10.1080/03610918.2021.1960373
5. Qasim M, Månsson K, Golam Kibria BM. On some beta ridge regression estimators: method, simulation and application. *J Stat Comput Simul.* (2021) 91:1699–712. doi: 10.1080/00949655.2020.1867549
6. Alobaidi NN, Shamany RE, Algamal ZY. A new ridge estimator for the negative binomial regression model. *Thailand Statistician.* (2021) 19:116–25.
7. Sami F, Amin M, Butt MM. On the ridge estimation of the Conway-Maxwell Poisson regression model with multicollinearity: Methods and applications. *Concurr Computat Pract Exp.* (2021) 2021:6477. doi: 10.1002/cpe.6477

8. Liu K. A new class of biased estimate in linear regression. *Commun Statist Theor Method.* (1993) 22:393–402. doi: 10.1080/03610929308831027
9. Stein C. Inadmissibility of the usual estimator for the mean of a multivariate normal distribution. In: *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. Berkeley, CA: University of California Press. (1956). p. 197–206. doi: 10.1525/9780520313880-018
10. Karlsson P, Månsson K, Kibria BG. A Liu estimator for the beta regression model and its application to chemical data. *J Chemom.* (2020) 34:e3300. doi: 10.1002/cem.3300
11. Qasim M, Kibria BMG, Månsson K, Sjölander P. A new Poisson Liu regression estimator: method and application. *J Appl Stat.* (2020) 47:2258–71. doi: 10.1080/02664763.2019.1707485
12. Naveed K, Amin M, Afzal S, Qasim M. New shrinkage parameters for the inverse Gaussian Liu regression. *Commun Statist Theor Method.* (2020) 2020:1–21. doi: 10.1080/03610926.2020.1791339
13. Liu K. Using Liu-type estimator to combat collinearity. *Commun Statist Theor Method.* (2003) 32:1009–20. doi: 10.1081/STA-120019959
14. Liu K. More on Liu-type estimator in linear regression. *Commun Statist Theor Method.* (2004) 33:2723–33. doi: 10.1081/STA-200037930
15. Özkale MR, Kaciranlar S. The restricted and unrestricted two-parameter estimators. *Commun Stat Theor Method.* (2007) 36:2707–25. doi: 10.1080/03610920701386877
16. Li Y, Yang H. A new Liu-type estimator in linear regression model. *Stat Pap.* (2012) 53:427–37. doi: 10.1007/s00362-010-0349-y
17. Kurnaz FS, Akay KU. A new Liu-type estimator. *Stat Pap.* (2015) 56:495–517. doi: 10.1007/s00362-014-0594-6
18. Sahriman S, Koerniawan V. Liu-type regression in statistical downscaling models for forecasting monthly rainfall salt as producer regions in Pangkep regency. *J Phys Conf Ser.* (2019) 1341:092021. doi: 10.1088/1742-6596/1341/9/092021
19. Algamal ZY, Abonazel MR. Developing a Liu-type estimator in beta regression model. *Concurr Computat Pract Exp.* (2021) 2021:e6685. doi: 10.1002/cpe.6685
20. Huang J, Yang H. A two-parameter estimator in the negative binomial regression model. *J Stat Comput Simul.* (2014) 84:124–34. doi: 10.1080/00949655.2012.696648
21. Algamal ZY. Shrinkage estimators for gamma regression model. *Electr J Appl Stat Anal.* (2018) 11:253–68. doi: 10.1285/i20705948v11n1p253
22. Asar Y, Genç A. A new two-parameter estimator for the Poisson regression model. *Iran J Sci Technol Trans A Sci.* (2018) 42:793–803. doi: 10.1007/s40995-017-0174-4
23. Rady EA, Abonazel MR, Taha IM. A new biased estimator for zero-inflated count regression models. *J Modern Appl Stat Method.* (2019). Available online at: https://www.researchgate.net/publication/337155202_A_New_Biased_Estimator_for_Zero-Inflated_Count_Regression_Models
24. Rady EA, Abonazel MR, Taha IM. New shrinkage parameters for Liu-type zero inflated negative binomial estimator. In: *The 54th Annual Conference on Statistics, Computer Science, and Operation Research 3-5 Dec, 2019*. Giza: FGSSR, Cairo University (2019).
25. Çetinkaya M, Kaçiranlar S. Improved two-parameter estimators for the negative binomial and Poisson regression models. *J Stat Comput Simul.* (2019) 89:2645–60. doi: 10.1080/00949655.2019.1628235
26. Abonazel MR, Farghali RA. Liu-type multinomial logistic estimator. *Sankhya B.* (2019) 81:203–25. doi: 10.1007/s13571-018-0171-4
27. Akram MN, Amin M, Qasim M. A new Liu-type estimator for the Inverse Gaussian Regression Model. *J Stat Comput Simul.* (2020) 90:1153–72. doi: 10.1080/00949655.2020.1718150
28. Lukman AF, Aladeitan B, Ayinde K, Abonazel MR. Modified ridge-type for the Poisson regression model: simulation and application. *J Appl Stat.* (2021) 2021:1–13. doi: 10.1080/02664763.2021.1889998
29. Ferrari S, Cribari-Neto F. Beta regression for modelling rates and proportions. *J Appl Stat.* (2004) 31:799–815. doi: 10.1080/0266476042000214501
30. Smithson M, Verkuilen J. A better lemon squeezer? Maximum-likelihood regression with beta-distributed dependent variables. *Psychol Method.* (2006) 11:54–71. doi: 10.1037/1082-989X.11.1.54
31. Cribari-Neto F, Zeileis A. Beta regression in R. *J Stat Softw.* (2010) 34:1–24. doi: 10.18637/jss.v034.i02
32. Bayer FM, Cribari-Neto F. Model selection criteria in beta regression with varying dispersion. *Commun Stat Simul Computat.* (2017) 46:729–46. doi: 10.1080/03610918.2014.977918
33. Espinheira PL, Ferrari SL, Cribari-Neto F. On beta regression residuals. *J Appl Stat.* (2008) 35:407–19. doi: 10.1080/02664760701834931
34. Espinheira PL, da Silva LCM, Silva ADO. Prediction measures in beta regression models. *arXiv preprint*. (2015). arXiv: 1501.04830.
35. Espinheira PL, da Silva LCM, Silva ADO, Ospina R. Model selection criteria on beta regression for machine learning. *Machine Learn Knowl Extr.* (2019) 1:427–49. doi: 10.3390/make1010026
36. Farebrother RW. Further results on the mean square error of ridge regression. *J Royal Stat Soc Ser B.* (1976) 38:248–50. doi: 10.1111/j.2517-6161.1976.tb01588.x
37. Hoerl AE, Kennard RW, Baldwin KF. Ridge regression: some simulations. *Commun Stat Theor Method.* (1975) 4:105–23. doi: 10.1080/03610917508548342
38. Kibria BMG. Performance of some new ridge regression estimators. *Commun Stat Simul Computat.* (2003) 32:419–35. doi: 10.1081/SAC-120017499
39. Farghali RA, Qasim M, Kibria BG, Abonazel MR. Generalized two-parameter estimators in the multinomial logit regression model: methods, simulation and application. *Commun Stat Simul Computat.* (2021) 1–16. doi: 10.1080/03610918.2021.1934023
40. Dawoud I, Abonazel MR. Robust Dawoud–Kibria estimator for handling multicollinearity and outliers in the linear regression model. *J Stat Comput Simul.* (2021) 91:3678–92. doi: 10.1080/00949655.2021.1945063
41. Awwad FA, Dawoud I, Abonazel MR. Development of robust Özkale-Kaçiranlar and Yang-Chang estimators for regression models in the presence of multicollinearity and outliers. *Concurr Computat Pract Exp*. (2021) 2021:e6779. doi: 10.1002/cpe.6779
42. Prater NH. Estimate gasoline yields from crude. *PetroLium Refiner.* (1956) 35:236–8.
43. Ospina R, Cribari-Neto F, Vasconcellos KL. Improved point and interval estimation for a beta regression model. *Comput Stat Data Anal.* (2006) 51:960–81. doi: 10.1016/j.csda.2005.10.002
44. Atkinson AC. *Plots, Transformations and Regression: An Introduction to Graphical Methods of Diagnostic Regression Analysis*. New York, NY: Oxford University Press (1985).
45. Arashi M, Norouzirad M, Roozbeh M, Mamode Khan N. A high-dimensional counterpart for the ridge estimator in multicollinear situations. *Mathematics.* (2021) 9:3057. doi: 10.3390/math9233057

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers in Applied Mathematics and Statistics

ORIGINAL RESEARCH
published: 22 February 2022
doi: 10.3389/fams.2022.775068

Dawoud–Kibria Estimator for Beta Regression Model: Simulation and Application

Mohamed R. Abonazel[1]*, *Issam Dawoud*[2], *Fuad A. Awwad*[3] *and Adewale F. Lukman*[4]

[1] Department of Applied Statistics and Econometrics, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza, Egypt, [2] Department of Mathematics, Al-Aqsa University, Gaza City, Palestine, [3] Department of Quantitative Analysis, College of Business Administration, King Saud University, Riyadh, Saudi Arabia, [4] Biostatistics and Epidemiology, University of Medical Sciences, Ondo City, Nigeria

The linear regression model becomes unsuitable when the response variable is expressed as percentages, proportions, and rates. The beta regression (BR) model is more appropriate for the variable of this form. The BR model uses the conventional maximum likelihood estimator (BML), and this estimator may not be efficient when the regressors are linearly dependent. The beta ridge estimator was suggested as an alternative to BML in the literature. In this study, we developed the Dawoud–Kibria estimator to handle multicollinearity in the BR model. The properties of the new estimator are derived. We compared the performance of the estimator with the existing estimators theoretically using the mean squared error criterion. A Monte Carlo simulation and a real-life application were carried out to show the benefits of the proposed estimator. The theoretical comparison, simulation, and real-life application results revealed the superiority of the proposed estimator.

Keywords: beta Kibria–Lukman estimator, beta Özkale–Kaçiranlar estimator, beta ridge estimator, maximum likelihood, mean square

OPEN ACCESS

Edited by:
Lixin Shen,
Syracuse University, United States

Reviewed by:
Xueying Zeng,
Ocean University of China, China
Erol Egrioglu,
Giresun University, Turkey
Zakariya Yahya Algamal,
University of Mosul, Iraq
Mahdi Roozbeh,
Semnan University, Iran

****Correspondence:***
Mohamed R. Abonazel
mabonazel@cu.edu.eg

Specialty section:
This article was submitted to Optimization,
a section of the journal
Frontiers in Applied Mathematics and Statistics

Received: *13 September 2021*
Accepted: *17 January 2022*
Published: *22 February 2022*

Citation:
Abonazel MR, Dawoud I, Awwad FA and Lukman AF (2022) Dawoud–Kibria Estimator for Beta Regression Model: Simulation and Application. Front. Appl. Math. Stat. 8:775068. doi: 10.3389/fams.2022.775068

INTRODUCTION

The linear regression (LR)model is used if the dependent variable follows a normal distribution. The assumption of the normality of the dependent variable may be violated and then it will fit some of the exponential family distributions as a negative binomial, Poisson, gamma, inverse Gaussian, and beta, so in this case, we use the generalized linear (GL) model instead of the LR model. The beta regression (BR) model is applied in many different fields such as engineering, medical sciences, physical sciences, social sciences, environment, and business if the dependent variable observations are between (0, 1). To estimate the BR model parameters, we use the maximum likelihood (ML) estimator which is more convenient than the ordinary least squares (OLS) estimator for describing and investigating different phenomena.

In the LR model, the explanatory variables may be correlated and this causes a problem called multicollinearity in which this problem may arise in the BR model. The ML estimator is the most popular used method for estimating the unknown regression parameters in the BR model. But also, in the existence of multicollinearity problems, the regression parameters' variances and standard errors are very large. To reduce the multicollinearity effect, different biased estimation methods are proposed and the most popular method is the ordinary ridge regression (ORR) estimation method which was proposed by Hoerl and Kennard [1, 2]. Another recent one parameter estimator proposed by Kibria and Lukman [3] to solve the multicollinearity is the Kibria and Lukman estimator. Also, in the case of an estimator with two parameters, Özkale and Kaçiranlar [4] proposed a two-parameter estimator. Very recently, Dawoud and Kibria [5] proposed a new kind of two-parameter estimator called the Dawoud–Kibria (DK) estimator. There are other recent studies regarding the one parameter and two-parameter estimators in LR and GL models, such as Roozbeh et al. [6], Lukman et al. [7], Arashi et al. [8], Farghali et al. [9], Lukman et al. [10, 11], Algamal and Abonazel [12], Akram et al. [13], and Abonazel et al. [14]. In this article, we drive the Dawoud–Kibria estimator for the BR model in the presence of the multicollinearity problem. Then, the properties of the Dawoud–Kibria estimator for the BR model are investigated.

This article is organized as follows. The methodology and the proposed estimator are given in section methodology. In section the superiority of the proposed estimator, the theoretical comparisons among the estimators are conducted. Section selection of biasing parameters k and d gives the proposed biasing parameters for the estimators. In sections Monte Carlo simulation study and real data application, the Monte Carlo simulation and the real-life dataset results are presented. Finally, in section conclusion, some conclusions of this article are given.

METHODOLOGY

In this section, we discuss the BR model. Then, the ridge, Kibria–Lukman, and Özkale–Kaçiranlar estimators are stated to the BR model. After that, we introduce the Dawoud–Kibria estimator for the BR model. Finally, the biasing parameters of the Dawoud–Kibria estimator for the BR model are proposed.

The BR Model

The BR model is popularly used in many different fields such as economics and medical studies. The BR model is used to show the effect of explanatory variables on a non-normal response variable as any generalized LR model. However, the response variable for the BR model is restricted to the interval (0, 1) as rates, proportions, and fractions. The BR model was given firstly by the authors Ferrari and Cribari-Neto [15] with relating the response variable mean function to linear predictors set through a link function. The BR model has a precision parameter where its reciprocal is determined as a dispersion measure [16, 17].

Let y be a continuous random variable having a beta distribution, then the probability density function of y is given as:

$$\mathrm{f}\left(y;\mu,\phi\right)=\frac{\Gamma\left(\phi\right)}{\Gamma\left(\mu\phi\right)\Gamma\left(\left(1-\mu\right)\phi\right)}y^{\mu\phi-1}\left(1-y\right)^{(1-\mu)\phi-1};\quad 0<y<1, 0<\mu<1, \phi>0, \tag{1}$$

where $\Gamma(\cdot)$ is called as the gamma function and ϕ is called as the precision parameter. The beta probability distribution mean and variance are as follows:

$$\mathrm{E}\left(y\right)=\mu, \mathrm{Var}\left(y\right)=\frac{Var\left(\mu\right)}{1+\phi}=\frac{\mu\left(1-\mu\right)}{1+\phi}.$$

Let $y_1,\ldots,y_n$ be independent random variables, where each y_i; $i=1,\ldots,n$ follows the density in Equation (1) with mean μ_i and unknown precision ϕ. The model is obtained by assuming that the mean of y_i can be written as:

$$\mathrm{g}\left(\mu_i\right)=\log\left(\frac{\mu_i}{1-\mu_i}\right)=x_i{}'\beta=\eta_i, \tag{2}$$

where $\mathrm{g}(\cdot)$ is the used link function, $\beta=\left(\beta_1,\ldots,\beta_p\right)'$ is an $\left(p\times 1\right)$ unknown parameters vector, $x_i=\left(x_{i1},\ldots,x_{ip}\right)'$ is the vector of p regressors, and η_i is the linear predictor.

Beta Maximum Likelihood Estimator

The BR parameters estimation is done using the beta maximum likelihood (BML) method [18]. The BR log-likelihood function is given as:

$$\mathcal{L}\left(\beta\right)=\sum_{i=1}^{n}\left\{\log\Gamma\left(\phi\right)-\log\Gamma\left(\mu_i\phi\right)-\log\Gamma\left(\left(1-\mu_i\right)\phi\right)+\left(\mu_i\phi-1\right)\log\left(y_i\right)+\left(\left(1-\mu_i\right)\phi-1\right)\log\left(1-y_i\right)\right\}. \tag{3}$$

Differentiating the log-likelihood given in Equation (3) with respect to the parameter β provides us the score function of the parameter β that is given as:

$$U\left(\beta\right)=\phi X'T\left(y^*-\mu^*\right), \tag{4}$$

where $T=\mathrm{diag}\left(\frac{1}{\mathrm{g}'(\mu_1)},\ldots,\frac{1}{\mathrm{g}'(\mu_n)}\right)$; with $\mathrm{g}'(\cdot)$ is the first derivative of $\mathrm{g}(\cdot)$; with $y_i^*=\log\left(\frac{y_i}{1-y_i}\right)$, and $\mu^*=\left(\mu_1^*,\ldots\mu_n^*\right)'$; with $\mu_i^*=\psi\left(\mu_i\phi\right)-\psi\left(\left(1-\mu_i\right)\phi\right)$, such that $\psi(\cdot)$ denoting the digamma function. The iterative reweighted least-squares (IRLS) algorithm or the Fisher scoring algorithm are used for estimating the parameter β [19, 20]. This algorithm form is given as:

$$\beta^{(r+1)}=\beta^{(r)}+\left(I_{\beta\beta}^{(r)}\right)^{-1}U_\beta^{(r)}\left(\beta\right), \tag{5}$$

where $U_\beta^{(r)}$ is called the score function, and $I_{\beta\beta}^{(r)}$ is called the information matrix for β, for more details, see Espinheira et al. [20]. With the use of the IRLS algorithm with initial values of β and ϕ as in Ferrari and Cribari-Neto [15] and Espinheira et al. [20], the BML estimator of the parameter β is provided as:

$$\hat{\beta}_{\mathrm{BML}}=\left(X'\hat{W}X\right)^{-1}X'\hat{W}z, \tag{6}$$

where X is an $\left(n\times p\right)$ design matrix, $z=\hat{\eta}+\hat{W}^{-1}\hat{T}\left(y^*-\hat{\mu}^*\right)$, and $\hat{W}=\mathrm{diag}\left(\hat{w}_1,\ldots,\hat{w}_n\right)$; with

$$\hat{w}_i=\hat{\phi}\left\{\psi'\left(\hat{\mu}_i\hat{\phi}\right)+\psi'\left(\left(1-\hat{\mu}_i\right)\hat{\phi}\right)\right\}\frac{1}{\left[\mathrm{g}'\left(\hat{\mu}_i\right)\right]^2}.$$

Here, $\hat{W}$, $\hat{T}$, $\hat{\mu}_i$, and $\hat{\mu}^*$ are the estimates of W, T, μ_i, and μ^*, respectively, evaluated at the ML estimator of β and ϕ [15].

Now, let $\Gamma=\mathrm{diag}\left(\gamma_1,\ldots,\gamma_p\right)=Q'X'\hat{W}XQ$, and $\alpha=\left(\alpha_1,\ldots,\alpha_p\right)'=Q'\beta$; where $\gamma_1\geq\ldots\geq\gamma_p\geq 0$ and Q is the matrix whose columns are the eigenvectors of the $\left(X'\hat{W}X\right)$ matrix. Then, the mean squared error matrix (MSEM) and the mean squared error (MSE) of an estimator $\tilde{\beta}$ are defined as follows:

$$MSEM(\tilde{\beta}) = Var(\tilde{\beta}) + \left(Bias(\tilde{\beta})\right)\left(Bias(\tilde{\beta})\right)', \tag{7}$$

$$MSE(\tilde{\beta}) = trace\left(MSEM(\tilde{\beta})\right). \tag{8}$$

Then the MSEM and MSE of $\hat{\beta}_{BML}$ are.

$$MSEM(\hat{\beta}_{BML}) = \frac{1}{\phi}\Gamma^{-1}, \tag{9}$$

$$MSE(\hat{\beta}_{BML}) = \frac{1}{\phi}\sum_{j=1}^{p}\frac{1}{\gamma_j}. \tag{10}$$

Beta Ridge Regression (BRR) Estimator

To reduce the effects of multicollinearity in the BR model, Abonazel and Taha [21] and Qasim et al. [22] introduced the BRR estimator as an alternative to the BML estimator and is given as:

$$\hat{\beta}_{BRR} = \left(X'\hat{W}X + k\,\mathrm{I}_p\right)^{-1}X'\hat{W}z, \ldots k > 0. \tag{11}$$

The MSEM and MSE of $\hat{\beta}_{BRR}$ are

$$\begin{aligned} MSEM(\hat{\beta}_{BRR}) &= \frac{1}{\phi}UL^{-1}\Gamma L^{-1}U' \\ &+ (UL^{-1}\Gamma U' - I_p)\alpha\alpha'(UL^{-1}\Gamma U' - I_p)', \end{aligned} \tag{12}$$

$$MSE(\hat{\beta}_{BRR}) = \frac{1}{\phi}\sum_{j=1}^{p}\frac{\gamma_j}{L_j^2} + k^2\sum_{j=1}^{p}\frac{\alpha_j^2}{L_j^2} \tag{13}$$

where $L = (\Gamma + k\,I_p)$ and $L_j = (\gamma_j + k)$.

Beta Kibria–Lukman (BKL) Estimator

The BKL estimator is defined as follows:

$$\hat{\beta}_{BKL} = \left(X'\hat{W}X + k\,\mathrm{I}_p\right)^{-1}\left(X'\hat{W}X - k\,\mathrm{I}_p\right)\hat{\beta}_{BML}, \quad k > 0. \tag{14}$$

The MSEM and MSE of $\hat{\beta}_{BKL}$ are

$$\begin{aligned} MSEM(\hat{\beta}_{BKL}) &= \frac{1}{\phi}UL^{-1}N\Gamma^{-1}NL^{-1}U' \\ &+ (UL^{-1}NU' - I_p)\alpha\alpha'(UL^{-1}NU' - I_p)', \end{aligned} \tag{15}$$

$$MSE(\hat{\beta}_{BKL}) = \frac{1}{\phi}\sum_{j=1}^{p}\frac{N_j^2}{\gamma_j L_j^2} + 4k^2\sum_{j=1}^{p}\frac{\alpha_j^2}{L_j^2} \tag{16}$$

where $N = (\Gamma - k\,I_p)$ and $N_j = (\gamma_j - k)$.

TABLE 1 | Simulated mean square error (SMSE) values of different estimators when $p = 2$ and $\phi = 2$.

n	ρ	BML	BRR	BKL	BOK	BDK(1)	BDK(2)
50	0.80	6.053	5.233	5.202	4.891	3.394	3.677
	0.85	7.303	6.118	6.069	5.503	3.395	3.838
	0.90	13.692	11.901	11.848	10.107	4.656	6.070
	0.95	31.207	26.815	26.691	19.919	4.923	8.081
	0.99	67.136	47.857	46.909	24.971	15.684	7.492
75	0.80	5.348	4.832	4.816	4.652	3.656	3.785
	0.85	7.008	6.210	6.185	5.792	3.933	4.325
	0.90	10.863	9.611	9.577	8.573	4.808	5.736
	0.95	18.291	14.916	14.788	11.584	3.846	5.274
	0.99	68.451	53.912	53.360	31.408	8.463	5.586
100	0.80	3.933	3.621	3.609	3.566	3.152	3.153
	0.85	9.107	8.289	8.271	7.700	5.047	5.730
	0.90	9.991	8.846	8.815	8.019	4.732	5.544
	0.95	15.744	13.466	13.396	11.168	4.514	6.142
	0.99	115.376	102.758	102.521	65.853	5.542	14.038
150	0.80	6.437	6.100	6.095	5.940	4.889	5.030
	0.85	6.972	6.518	6.510	6.286	4.929	5.129
	0.90	10.034	9.210	9.195	8.569	5.607	6.456
	0.95	18.945	17.151	17.119	14.781	6.782	9.339
	0.99	115.789	106.225	106.100	73.332	8.145	23.265
200	0.80	5.511	5.243	5.239	5.150	4.441	4.421
	0.85	6.501	6.162	6.157	5.999	4.966	5.051
	0.90	8.751	8.133	8.123	7.712	5.521	6.094
	0.95	16.097	14.810	14.791	13.240	7.235	9.253
	0.99	146.696	138.772	138.709	102.489	19.521	45.366

Beta Özkale–Kaçiranlar (BOK) Estimator

Recently, Abonazel et al. [14] proposed the BOK estimator as an extension of the Özkale and Kaçiranlar [4] estimator in the BR model and is defined as follows:

$$\hat{\beta}_{BOK} = (X'\,\hat{W}X + k\,\mathrm{I}_p)^{-1}(X'\hat{W}\,X + kd\mathrm{I}_p)\,\hat{\beta}_{BML}, \quad k > 0, \quad 0 < d < 1. \qquad (17)$$

The MSEM and MSE of $\hat{\beta}_{BOK}$ are

$$MSEM(\hat{\beta}_{BOK}) = \frac{1}{\phi}UL^{-1}G\Gamma^{-1}GL^{-1}U' + (UL^{-1}GU' - I_p)\alpha\alpha'(UL^{-1}GU' - I_p)', \qquad (18)$$

$$MSE(\hat{\beta}_{BOK}) = \frac{1}{\phi}\sum_{j=1}^{p}\frac{G_j^2}{\gamma_j L_j^2} + (1-d)^2k^2\sum_{j=1}^{p}\frac{\alpha_j^2}{L_j^2} \qquad (19)$$

where $G = (\Gamma + kd\,I_p)$ and $G_j = (\gamma_j + kd)$.

The Proposed Estimator

Extensions of the two-parameter estimators to the area of GLMs have been recently developed; such as Qasim et al. [22], Farghali et al. [9], Lukman et al. [23], Algamal and Abonazel [12], and Abonazel et al. [14]. Following the previous works, we introduced the beta version of the two-parameter estimator of Dawoud and Kibria [5] (BDK) as follows:

$$\hat{\beta}_{BDK} = (X'\,\hat{W}\,X + k(1+d)\mathrm{I}_p)^{-1}(X'\,\hat{W}X - k\,(1+d)\mathrm{I}_p)\hat{\beta}_{BML}, \quad k > 0, \;\; 0 < d < 1. \qquad (20)$$

We give the MSEM of the proposed $\hat{\beta}_{BDK}$ as follows:

$$MSEM(\hat{\beta}_{BDK}) = \frac{1}{\phi}UM^{-1}R\Gamma^{-1}RM^{-1}U' + (UM^{-1}RU' - \mathrm{I}_p)\alpha\alpha'(UM^{-1}RU' - I_p)', \qquad (21)$$

$$MSE(\hat{\beta}_{BDK}) = \frac{1}{\phi}\sum_{j=1}^{p}\frac{R_j^2}{\gamma_j M_j^2} + 4k^2(1+d)^2\sum_{j=1}^{p}\frac{\alpha_j^2}{M_j^2}, \qquad (22)$$

where $M = (\Gamma + k(1+d)\,I_p)$, $R = (\Gamma - k(1+d)\,I_p)$, $M_j = (\gamma_j + k(1+d))$ and $R_j = (\gamma_j - k(1+d))$.

THE SUPERIORITY OF THE PROPOSED ESTIMATOR

Theorem 1: If $4k^2(1+d)^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2 < \sum_{j=1}^{p}\left(M_j^2 - R_j^2\right)$, then $MSE\,(\hat{\beta}_{BDK}) < MSE\,(\hat{\beta}_{BML})$.

TABLE 2 | SMSE values of different estimators when $p = 2$ and $\phi = 6$.

n	ρ	BML	BRR	BKL	BOK	BDK(1)	BDK(2)
50	0.80	11.491	10.645	10.627	9.735	7.130	6.903
	0.85	13.247	12.228	12.206	11.035	7.716	7.654
	0.90	18.158	16.463	16.425	14.042	7.790	8.644
	0.95	35.625	32.457	32.400	25.909	10.109	13.399
	0.99	192.874	168.676	168.149	93.746	17.848	12.119
75	0.80	15.037	14.280	14.271	13.177	9.543	9.539
	0.85	17.553	16.759	16.751	15.364	10.642	11.255
	0.90	27.598	26.653	26.646	24.351	16.533	18.294
	0.95	59.231	56.845	56.829	47.709	21.969	30.398
	0.99	214.373	186.262	185.683	103.517	16.966	10.659
100	0.80	11.001	10.576	10.571	10.034	8.496	7.578
	0.85	14.828	14.055	14.046	13.017	9.301	9.558
	0.90	19.577	18.642	18.633	17.090	11.662	12.394
	0.95	43.621	41.238	41.215	34.604	15.749	21.251
	0.99	265.201	252.353	252.261	175.379	22.736	61.884
150	0.80	10.789	10.545	10.544	10.227	9.133	8.370
	0.85	12.325	11.800	11.794	11.136	8.899	8.413
	0.90	17.602	16.783	16.775	15.534	10.948	11.554
	0.95	46.605	45.329	45.324	40.763	25.761	31.100
	0.99	252.047	245.921	245.900	195.177	67.404	114.900
200	0.80	9.358	9.133	9.131	8.897	8.099	7.276
	0.85	15.565	15.263	15.262	14.763	12.608	12.707
	0.90	21.999	21.571	21.569	20.598	16.733	17.630
	0.95	28.486	27.156	27.146	24.343	14.704	17.431
	0.99	213.727	207.448	207.422	163.383	53.122	92.802

Proof: The MSE difference between the BML and the BDK estimators is written as

$$\Delta_1 = MSE\left(\hat{\beta}_{BDK}\right) - MSE\left(\hat{\beta}_{BML}\right) \\ = \frac{1}{\phi}\sum_{j=1}^{p}\left[\frac{R_j^2 - M_j^2 + 4k^2(1+d)^2\gamma_j\phi\alpha_j^2}{\gamma_j M_j^2}\right]. \qquad (23)$$

In the case of $R_j^2 - M_j^2 + 4k^2(1+d)^2\gamma_j\phi\alpha_j^2 < 0$ in the equation (23), it implies that $4k^2(1+d)^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2 < \sum_{j=1}^{p}\left(M_j^2 - R_j^2\right)$, then $MSE\left(\hat{\beta}_{BDK}\right) < MSE\left(\hat{\beta}_{BML}\right)$. That means the BDK estimator is better than the BML estimator if $4k^2(1+d)^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2 < \sum_{j=1}^{p}\left(M_j^2 - R_j^2\right)$.

Theorem 2: If $\sum_{j=1}^{p}\left(R_j^2L_j^2 - \gamma_j^2M_j^2\right) < k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2(M_j^2 - 4(1+d)^2L_j^2)$, then $MSE\left(\hat{\beta}_{BDK}\right) < MSE\left(\hat{\beta}_{BRR}\right)$.

Proof: The MSE difference between the BRR and the BDK estimators is written as

$$\Delta_2 = MSE\left(\hat{\beta}_{BDK}\right) - MSE\left(\hat{\beta}_{BRR}\right) \\ = \frac{1}{\phi}\sum_{j=1}^{p}\left[\frac{R_j^2L_j^2 - \gamma_j^2M_j^2 - k^2\phi\gamma_j\alpha_j^2(M_j^2 - 4(1+d)^2L_j^2)}{\gamma_j L_j^2 M_j^2}\right]. \qquad (24)$$

In the case of $R_j^2L_j^2 - \gamma_j^2M_j^2 - k^2\phi\gamma_j\alpha_j^2(M_j^2 - 4(1+d)^2L_j^2) < 0$ in the Equation (24), it implies that $\sum_{j=1}^{p}\left(R_j^2L_j^2 - \gamma_j^2M_j^2\right) < k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2(M_j^2 - 4(1+d)^2L_j^2)$, then $MSE\left(\hat{\beta}_{BDK}\right) < MSE\left(\hat{\beta}_{BRR}\right)$. That means the BDK estimator is better than the BRR estimator if $\sum_{j=1}^{p}\left(R_j^2L_j^2 - \gamma_j^2M_j^2\right) < k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2(M_j^2 - 4(1+d)^2L_j^2)$.

Theorem 3: If $\sum_{j=1}^{p}\left(R_j^2L_j^2 - N_j^2M_j^2\right) < 4k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2(M_j^2 - (1+d)^2L_j^2)$.

then $MSE\left(\hat{\beta}_{BDK}\right) < MSE\left(\hat{\beta}_{BKL}\right)$.

TABLE 3 | SMSE values of different estimators when $p = 4$ and $\phi = 2$.

n	ρ	BML	BRR	BKL	BOK	BDK(1)	BDK(2)
50	0.80	8.618	7.044	6.999	6.275	3.469	3.367
	0.85	9.465	7.472	7.431	6.421	3.357	3.368
	0.90	16.331	13.295	13.254	10.640	4.493	4.428
	0.95	35.235	29.336	29.247	20.055	5.494	5.071
	0.99	271.828	224.818	223.712	92.905	25.607	41.613
75	0.80	8.023	7.017	7.003	6.611	4.308	4.009
	0.85	10.160	8.556	8.535	7.630	4.303	4.124
	0.90	17.399	14.616	14.573	12.058	5.093	4.947
	0.95	31.134	25.589	25.472	18.173	4.529	4.793
	0.99	187.813	161.921	161.693	81.603	6.541	17.808
100	0.80	7.523	6.513	6.494	6.149	4.096	4.124
	0.85	9.167	8.000	7.983	7.366	4.665	4.648
	0.90	19.593	17.076	17.026	14.578	6.429	6.697
	0.95	31.651	27.423	27.365	21.103	6.951	6.707
	0.99	217.675	194.858	194.677	112.730	6.734	13.765
150	0.80	7.066	6.454	6.448	6.285	4.672	4.599
	0.85	9.429	8.514	8.504	8.044	5.535	5.626
	0.90	14.144	12.871	12.861	11.751	7.393	7.420
	0.95	33.151	29.960	29.924	24.822	10.332	10.585
	0.99	178.793	161.444	161.275	101.082	9.202	10.705
200	0.80	7.135	6.584	6.578	6.434	4.908	5.053
	0.85	7.726	7.044	7.036	6.819	4.988	5.190
	0.90	13.439	12.248	12.236	11.275	7.226	7.363
	0.95	30.354	28.056	28.043	24.286	12.687	12.634
	0.99	201.543	188.253	188.153	133.641	21.328	24.457

Proof: The MSE difference between the BKL and the BDK estimators is written as

$$\begin{aligned}\Delta_3 &= MSE\left(\hat{\beta}_{BDK}\right) - MSE\left(\hat{\beta}_{BKL}\right)\\ &= \frac{1}{\phi}\sum_{j=1}^{p}\left[\frac{R_j^2L_j^2 - N_j^2M_j^2 - 4k^2\phi\gamma_j\alpha_j^2\left(M_j^2-(1+d)^2L_j^2\right)}{\gamma_jL_j^2M_j^2}\right].\end{aligned} \qquad (25)$$

In the case of $R_j^2L_j^2 - N_j^2M_j^2 - 4k^2\phi\gamma_j\alpha_j^2(M_j^2-(1+d)^2L_j^2) < 0$ in the Equation (25), it implies that $\sum_{j=1}^{p}\left(R_j^2L_j^2 - N_j^2M_j^2\right) < 4k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2(M_j^2-(1+d)^2L_j^2)$, then $MSE\left(\hat{\beta}_{BDK}\right) < MSE\left(\hat{\beta}_{BKL}\right)$. That means the BDK estimator is better than the BKL estimator if $\sum_{j=1}^{p}\left(R_j^2L_j^2 - N_j^2M_j^2\right) < 4k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2(M_j^2-(1+d)^2L_j^2)$.

Theorem 4: If $\sum_{j=1}^{p}\left(R_j^2L_j^2 - G_j^2M_j^2\right) < k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2((1-d)^2M_j^2-4(1+d)^2L_j^2)$, then $MSE\left(\hat{\beta}_{BDK}\right) < MSE\left(\hat{\beta}_{BOK}\right)$.

Proof: The MSE difference between the BOK and the BDK estimators is written as

$$\begin{aligned}\Delta_4 &= MSE\left(\hat{\beta}_{BDK}\right) - MSE\left(\hat{\beta}_{BOK}\right)\\ &= \frac{1}{\phi}\sum_{j=1}^{p}\left[\frac{R_j^2L_j^2 - G_j^2M_j^2 - k^2\phi\gamma_j\alpha_j^2\left((1-d)^2M_j^2-4(1+d)^2L_j^2\right)}{\gamma_jL_j^2M_j^2}\right].\end{aligned} \qquad (26)$$

In the case of $R_j^2L_j^2 - G_j^2M_j^2 - k^2\phi\gamma_j\alpha_j^2((1-d)^2M_j^2 - 4(1+d)^2L_j^2) < 0$ in the Equation (26), it implies that $\sum_{j=1}^{p}\left(R_j^2L_j^2 - G_j^2M_j^2\right) < k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2((1-d)^2M_j^2-4(1+d)^2L_j^2)$, then $MSE\left(\hat{\beta}_{BDK}\right) < MSE\left(\hat{\beta}_{BOK}\right)$. That means the BDK estimator is better than the BOK estimator if $\sum_{j=1}^{p}\left(R_j^2L_j^2 - G_j^2M_j^2\right) < k^2\phi\sum_{j=1}^{p}\gamma_j\alpha_j^2((1-d)^2M_j^2-4(1+d)^2L_j^2)$.

SELECTION OF BIASING PARAMETERS *k* and *d*

We will suggest the following biasing parameters' estimators for the mentioned estimators.

TABLE 4 | SMSE values of different estimators when $p = 4$ and $\phi = 6$.

n	*ρ*	BML	BRR	BKL	BOK	BDK(1)	BDK(2)
50	0.80	20.276	19.341	19.333	17.555	11.620	11.975
	0.85	18.047	17.189	17.185	15.592	10.837	10.841
	0.90	27.687	25.923	25.917	22.257	12.368	12.520
	0.95	87.147	83.888	83.879	69.421	30.856	31.332
	0.99	603.570	577.427	577.263	357.585	36.519	43.205
75	0.80	14.407	13.659	13.653	12.829	9.262	9.390
	0.85	25.218	23.940	23.934	21.759	13.710	13.419
	0.90	38.729	37.702	37.702	34.643	24.505	26.369
	0.95	84.082	80.781	80.767	67.356	30.970	35.892
	0.99	472.368	460.962	460.940	338.777	90.885	87.179
100	0.80	16.825	16.274	16.272	15.473	11.865	13.190
	0.85	24.909	23.521	23.510	21.343	12.895	15.321
	0.90	30.072	28.630	28.622	25.641	15.512	17.823
	0.95	67.417	64.185	64.170	53.419	24.947	27.241
	0.99	658.305	644.268	644.236	479.563	123.607	135.271
150	0.80	13.969	13.573	13.572	13.099	10.726	11.658
	0.85	20.345	19.764	19.762	18.816	14.670	16.154
	0.90	33.991	33.263	33.261	31.344	23.972	26.417
	0.95	86.685	85.104	85.101	77.562	52.626	58.964
	0.99	535.740	526.202	526.187	422.009	160.031	192.949
200	0.80	17.816	17.532	17.532	17.071	14.878	15.815
	0.85	21.343	20.971	20.971	20.283	17.201	18.443
	0.90	36.550	35.967	35.966	34.245	27.566	29.896
	0.95	98.462	96.739	96.736	88.492	60.933	68.280
	0.99	516.039	507.722	507.706	413.974	171.268	199.438

Following Hoerl et al. [24] and Qasim et al. [22], $\hat{k}$ of the BRR estimator is written as

$$\hat{k}_{BRR} = \frac{p}{\hat{\phi}\sum_{j=1}^{p}\hat{\alpha}_j^2}, \tag{27}$$

where $\hat{\alpha}_j$ is the jth element of $\hat{\alpha} = Q'\hat{\beta}_{\mathrm{BML}}$ vector and $\hat{\phi}$ is the ML estimate of ϕ [15].

- Following Lukman et al. [25], $\hat{k}_{BKL}$ of the BKL estimator is written as

$$\hat{k}_{BKL} = \frac{p}{\hat{\phi}\sum_{j=1}^{p}\left(\frac{1}{\hat{\phi}\gamma_j}+2\hat{\alpha}_j^2\right)} \tag{28}$$

- Following Özkale and Kaçiranlar [4] and Abonazel et al. [14], $\hat{k}_{BOK}$ and $\hat{d}_{BOK}$ of the BOK estimator are written as

$$\hat{d}_{BOK} = \min\left(\frac{\hat{\alpha}_j^2}{\frac{1}{\hat{\phi}\gamma_j}+\hat{\alpha}_j^2}\right)_{j=1}^{p} \tag{29}$$

$$\hat{k}_{BOK} = \left(\frac{p}{\hat{\phi}\sum_{j=1}^{p}\left(\hat{\alpha}_j^2-\hat{d}_{BOK}\left(\frac{1}{\hat{\phi}\gamma_j}+\hat{\alpha}_j^2\right)\right)}\right)^{1/2} \tag{30}$$

- Following Dawoud and Kibria [5], we suggest two different $\hat{k}$ of the proposed BDK estimator as follows:

$$\hat{k}_{BDK(1)} = \left(\hat{k}_{BRR}\right)^{1/p} \tag{31}$$

$$\hat{k}_{BDK(2)} = \left(\frac{1}{p}\sum_{j=1}^{p}\frac{1}{\hat{\phi}\left(1+\hat{d}_{BOK}\right)\left(\frac{1}{\hat{\phi}\gamma_j}+2\hat{\alpha}_j^2\right)}\right)^{1/p} \tag{32}$$

MONTE CARLO SIMULATION STUDY

In this section, a Monte Carlo simulation study has been conducted to compare the performances of BML, BRR, BKL, and

TABLE 5 | SMSE values of different estimators when $p = 6$ and $\phi = 2$.

n	ρ	BML	BRR	BKL	BOK	BDK(1)	BDK(2)
50	0.80	11.105	9.078	9.057	7.810	4.655	3.876
	0.85	13.514	10.799	10.772	8.788	4.607	3.753
	0.90	25.321	20.354	20.247	14.685	5.540	4.364
	0.95	47.497	37.538	37.399	22.139	5.574	5.085
	0.99	264.803	213.947	212.957	76.134	20.147	57.276
75	0.80	7.189	5.922	5.914	5.500	3.811	3.354
	0.85	13.325	10.910	10.870	9.195	5.001	3.725
	0.90	19.350	16.098	16.061	12.725	6.394	4.503
	0.95	48.120	38.549	38.286	23.752	6.202	5.768
	0.99	243.478	204.310	203.653	83.874	10.836	41.688
100	0.80	9.013	7.461	7.427	6.761	4.419	3.543
	0.85	10.948	9.292	9.270	8.258	5.295	4.066
	0.90	16.251	13.651	13.624	11.390	6.264	4.555
	0.95	30.617	25.870	25.833	19.041	8.188	5.088
	0.99	231.060	201.644	201.293	96.261	8.733	20.457
150	0.80	7.345	6.380	6.365	6.074	4.492	3.810
	0.85	9.407	8.216	8.201	7.613	5.405	4.359
	0.90	15.455	13.587	13.566	11.983	7.540	5.237
	0.95	36.480	32.601	32.573	25.971	13.137	7.642
	0.99	228.210	207.108	206.955	119.693	20.897	11.495
200	0.80	7.730	6.957	6.950	6.700	5.232	4.412
	0.85	9.778	8.752	8.741	8.222	6.035	4.953
	0.90	12.742	11.311	11.302	10.195	7.161	5.536
	0.95	33.667	30.140	30.105	24.579	12.452	7.475
	0.99	219.889	204.152	204.074	131.491	37.952	10.424

BOK with the suggested estimator (BDK). The program of the simulation study is written in R programming language based on the *betareg* package.

The Design of the Experiment

We simulated the datasets with the following settings:

1) The response variable y_i is generated from the beta distribution as *Beta* $(\mu_i, \ \phi)$, where $\mu_i = \exp\left(x_i{}'\beta\right) / \left(1 + \exp\left(x_i{}'\beta\right)\right); \ i = 1, \ldots, n$, and x_i is the ith row of X. The precision parameter ϕ chosen in the simulation is $\phi = 2$ and 6.
2) Sample size: $n = 50$, 75, 100, 150, and 200.
3) Explanatory variables are generated with a degree of multicollinearity as in Kibria [26]: $x_{ij} = u_{ij}\sqrt{1-\rho^2} + \rho u_{ip}$, where u_{ij} are the independent standard uniform pseudorandom numbers, and ρ is defined as the correlation between the explanatory variables, $\rho = 0.80$, 0.85, 0.90, 0.95, and 0.99.
4) The number of explanatory variables is $p = 2$, 4, and 6; with $\beta'\beta = 1$ and $\beta_1 = \ldots = \beta_p$, as per Kaçiranlar and Dawoud [27], Rady et al. [28], Abonazel and Farghali [29], Farghali et al. [9], Dawoud and Abonazel [30], and Awwad et al. [31].
5) We used the simulated MSE (SMSE) criterion for verification, which are computed as

$$SMSE\left(\hat{\beta}\right) = \frac{1}{5000}\sum_{l=1}^{5000}\left(\hat{\beta}_l - \beta\right)'\left(\hat{\beta}_l - \beta\right), \tag{33}$$

where $\hat{\beta}_l$ is the estimated value vector at the lth experiment of the simulation, β is the true parameter vector. The number of replications is 5,000.

Simulation Results

We have the following comments according to the simulation results in **Tables 1–6**: Obviously, from **Tables 1–6**, the proposed estimator possesses a smaller SMSE than the BML estimator and other estimators understudy for all sample sizes. For instance, from **Table 3**, when $\rho = 0.9$, $n = 50$, the SMSE of BML is 16.331 while the SMSE for other estimators is as follows: 13.295 (BRR), 13.254 (BKL), 10.640 (BOK), 4.493 (BDK(1)), and 4.428 (BDK(2)), respectively. Similarly, when the values of ϕ increase the SMSE also increases: from **Table 1**, when $\phi = 2$, $n = 100$ and $\rho = 0.99$, and **Table 2**, when $\phi = 6$, $n = 100$ and $\rho = 0.99$, the SMSE of BRR rises from 102.758 to 252.353. Also, it is evident that the SMSE values of all the estimators increased as the number of explanatory p increased. For the one-parameter shrinkage estimator, the BKL estimator consistently dominates the BRR estimator. For two-parameter shrinkage estimators, the BDK estimator dominates

TABLE 6 | SMSE values of different estimators when $p = 6$ and $\phi = 6$.

n	ρ	BML	BRR	BKL	BOK	BDK(1)	BDK(2)
50	0.80	24.308	23.396	23.396	21.226	16.249	12.890
	0.85	32.188	30.588	30.577	26.475	17.289	12.522
	0.90	49.485	46.799	46.789	38.084	21.315	13.021
	0.95	101.291	97.077	97.075	76.197	39.597	22.003
	0.99	693.505	670.736	670.667	410.533	90.039	37.843
75	0.80	24.844	23.949	23.944	21.907	16.472	13.086
	0.85	34.987	33.539	33.532	29.804	20.256	16.453
	0.90	47.139	44.266	44.241	36.283	19.387	12.870
	0.95	93.080	89.724	89.717	73.289	41.415	24.254
	0.99	754.197	735.126	735.082	496.580	123.219	36.798
100	0.80	17.825	17.177	17.175	16.127	12.728	11.537
	0.85	21.441	20.510	20.506	18.863	13.433	11.505
	0.90	49.162	47.478	47.473	42.140	27.865	22.938
	0.95	103.039	99.905	99.896	85.178	47.090	35.859
	0.99	634.956	619.106	619.078	444.692	140.691	59.963
150	0.80	17.301	16.801	16.800	16.101	13.051	13.019
	0.85	32.077	31.463	31.463	30.007	24.389	23.897
	0.90	42.623	41.614	41.612	38.705	28.606	27.190
	0.95	98.059	95.272	95.263	83.046	49.071	42.380
	0.99	724.422	714.279	714.269	583.521	270.776	188.774
200	0.80	19.214	18.795	18.794	18.181	15.345	15.624
	0.85	26.161	25.717	25.716	24.805	21.034	21.107
	0.90	47.018	46.298	46.297	44.058	35.750	35.533
	0.95	108.860	107.407	107.405	99.800	75.129	71.876
	0.99	652.981	645.213	645.206	547.813	293.726	233.134

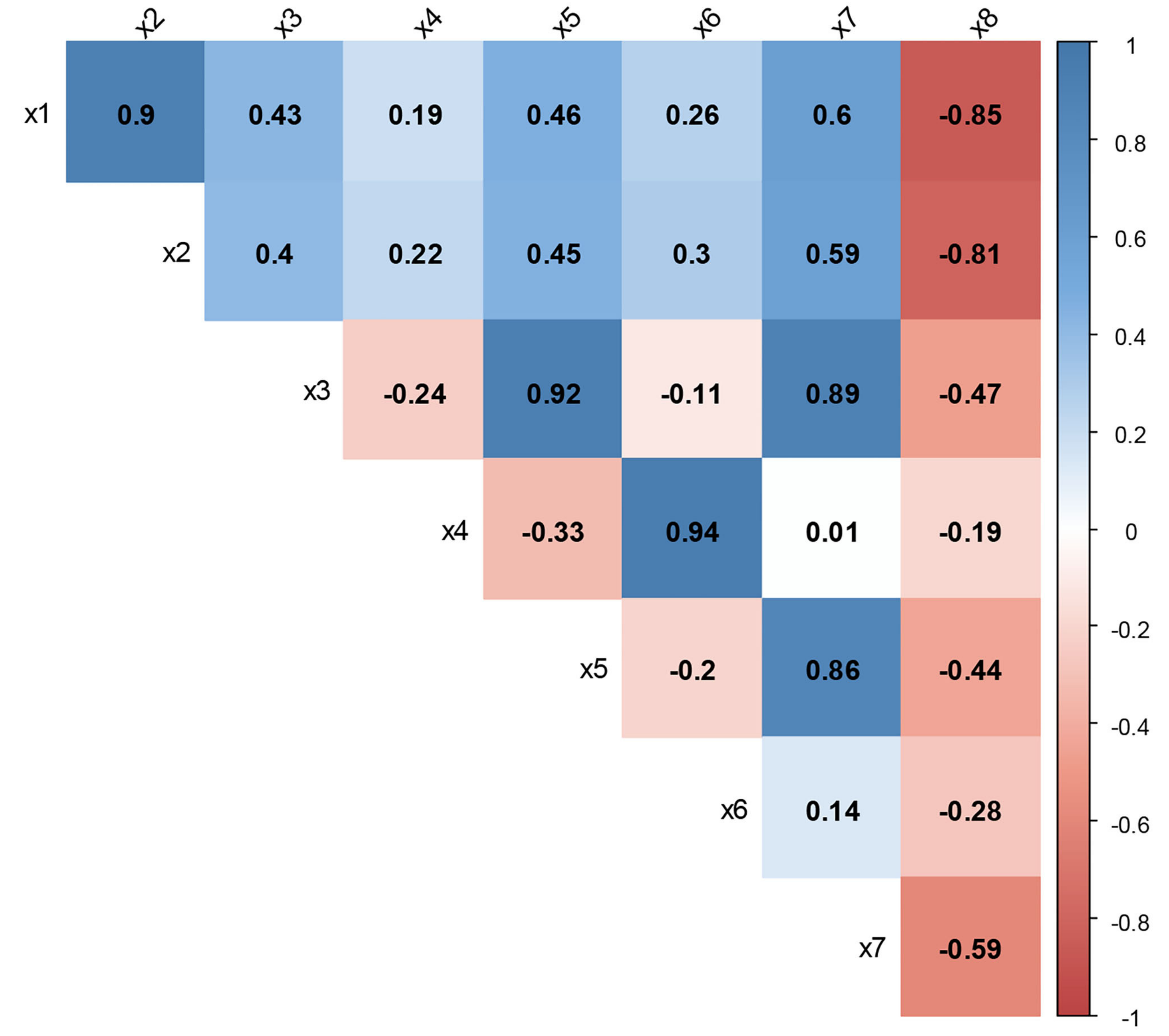


FIGURE 1 | Visualization of the correlation matrix.

the BOK estimator. Overall, the BDK dominates both the one-parameter and the two-parameter estimators. However, the performance of each estimator is a function of the employed shrinkage parameter.

REAL DATA APPLICATION

The implementation of the proposed estimator is illustrated by a study applied to the well-being index of Turkey in 2015 [32]. The index involves the aspects of accommodation, jobs, income and wealth, health, education, climate, protection, public engagement and access to community resources and social life. As the life satisfaction index is between 0 and 1. The values close to 1 refer to a better standard of living. The data are obtained from the Turkish Statistics Association. The original dataset consists of some dimensions that are represented by 41 indicators. Here, we are interested in only nine indicators used by Abonazel and Taha [21] and the number of observations is 50. The response variable is the level of happiness and eight explanatory variables are x1: Number of rooms per person, x2: Average point of necessary placement scores of the system for transition to secondary education from basic education, x3: Satisfaction rate with public education services, x4: Percentage of the population receiving waste services, x5: Satisfaction rate with public safety services, x6: The access rate of the population to sewerage and pipe system, x7: Satisfaction rate with public health services, and x8: Percentage of households declaring to fail on meeting basic needs.

TABLE 7 | Estimation results for the used estimators.

	BML	BRR	BKL	BOK	BDK(1)	BDK(2)
x1	−0.4269	−0.4028	−0.4022	−0.3942	−0.3584	0.2425
x2	0.0014	0.0012	0.0012	0.0012	0.0010	−0.0022
x3	0.0017	0.0018	0.0018	0.0018	0.0020	0.0048
x4	−0.0019	−0.0019	−0.0019	−0.0019	−0.0019	−0.0021
x5	−0.0076	−0.0076	−0.0076	−0.0077	−0.0077	−0.0085
x6	−0.0044	−0.0044	−0.0044	−0.0044	−0.0044	−0.0038
x7	0.0270	0.0269	0.0269	0.0269	0.0266	0.0229
x8	−0.0095	−0.0093	−0.0093	−0.0092	−0.0088	−0.0033
k	–	0.4997	0.2489	0.7182	0.7069	29.3690
d	–	–	–	0.0308	0.0308	0.0308
MSE	0.00138	0.00123	0.00122	0.00117	0.00097	0.00047
R^2	0.752	0.779	0.780	0.789	0.825	0.915
GCV	–	74.714	74.707	74.617	73.795	73.250

To investigate the multicollinearity through correlation coefficients between the explanatory variables, a visualization of the correlation matrix of the variables is constructed with the corresponding coefficients reported in **Figure 1**. The correlation coefficients indicate that there are strong relationships (more than 0.8) between some explanatory variables. This denotes the severe multicollinearity presence. Moreover, this conclusion is confirmed by the variance inflation factor (VIF) and the condition number $\left(\mathrm{CN} = \sqrt{\max(\gamma_j)/\min(\gamma_j)}\right)$ [33]; where the VIFs of the eight explanatory variables are 7.5, 6.1, 10.8, 10.1, 9.1, 9.8, 9.7, and 4.3, respectively, and the CN is 3,936.055.

Table 7 provides the regression parameter estimates for the BR model using BML, BRR, BKL, BOK, and BDK. From **Table 7**, it can note that the estimated regression parameters of all estimators have the same signs (except x1 and x2 in BDK(2) only); this means that the type of relationship between each explanatory variable and the response variable is not changed from what it was in the BML. The estimated MSE of the five estimators were obtained by Equations (10), (13), (16), (19), and (22), respectively. The results of **Table 7** indicate that the estimated MSE value of BML is greater than the estimated MSE values of BRR, BKL, BOK, and BDK estimators. Moreover, the MSE values of BDK(1) and BDK(2) estimators are lower than other estimators, which means that the BDK estimator achieves the best performance. Furthermore, in terms of the prediction, the R^2 value of the proposed estimator (BDK) is the greatest among all the used estimators. To further highlight the performance of the BDK estimator, generalized cross-validation (GCV) criterion is used in comparison [8, 34, 35]. Regarding GCV values, it can note that the BDK yielded the least value compared with other estimators.

Through this application, we verify the theoretical results as follows:

1. Since the condition $4\hat{k}^2_{BDK(2)}(1+\hat{d}_{BOK})^2\hat{\phi}\sum_{j=1}^{p}\gamma_j\hat{\alpha}_j^2 = 7.26e + 7 < \sum_{j=1}^{p}\left(\hat{M}_j^2 - \hat{R}_j^2\right) = 1.58e + 10$ is satisfied, then the BDK estimator is better than the BML estimator.
2. Since the condition $\sum_{j=1}^{p}\left(\hat{R}_j^2\hat{L}_j^2 - \gamma_j^2\hat{M}_j^2\right) = -1.35e + 26 < \hat{k}^2_{BDK(2)}\hat{\phi}\sum_{j=1}^{p}\gamma_j\hat{\alpha}_j^2(\hat{M}_j^2 - 4(1+\hat{d}_{BOK})^2\hat{L}_j^2) = -7.83e + 23$ is satisfied, then the BDK estimator is better than the BRR estimator.
3. Since the condition $\sum_{j=1}^{p}\left(\hat{R}_j^2\hat{L}_j^2 - \hat{N}_j^2\hat{M}_j^2\right) = -7.84e + 24 < 4\,\hat{k}^2_{BDK(2)}\,\hat{\phi}\sum_{j=1}^{p}\gamma_j\hat{\alpha}_j^2(\hat{M}_j^2 - (1+\hat{d}_{BOK})^2\hat{L}_j^2) = -6.03e + 22$ is satisfied, then the BDK estimator is better than the BKL estimator.
4. Since the condition $\sum_{j=1}^{p}\left(\hat{R}_j^2\hat{L}_j^2 - \hat{G}_j^2\hat{M}_j^2\right) = -1.39e + 26 < \hat{k}^2_{BDK(2)}\hat{\phi}\sum_{j=1}^{p}\gamma_j\hat{\alpha}_j^2((1-\hat{d}_{BOK})^2\hat{M}_j^2 - 4(1+\hat{d}_{BOK})^2\hat{L}_j^2) = -7.98e + 23$ is satisfied, then the BDK estimator is better than the BOK estimator.

CONCLUSION

Regression modeling describes the relationship that exists between a dependent variable and one or more explanatory variables. Linear dependency, a situation called multicollinearity, is a common problem with two or more explanatory variables. Multicollinearity is a threat to the efficiency of the maximum likelihood estimator in both the linear and generalized linear models, such as the BR model. The ridge regression estimator serves as an alternative to the maximum likelihood estimator for parameter estimation in the beta regression model. In this article, we developed the BDK estimator and compared its performance theoretically with some other estimators. A simulation study has been conducted to compare the performance of the estimators. Real-life data have been analyzed to illustrate the findings of the article. We concluded that the BDK estimator proposed in this articles generally preferred when there is multicollinearity in the beta regression model. For future work, for example, one can use new methods to select the shrinkage parameters as an extension to Uslu et al. [36] and Inan et al. [37] in the BR model, or provide robust biased estimators for handling multicollinearity and outliers together in the beta regression model as an extension to Awwad et al. [31] and Dawoud and Abonazel [30].

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

MA, ID, and FA contributed to conception and structural design of the manuscript. MA performed the simulation and application sections. AL wrote the abstract and conclusion sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

ACKNOWLEDGMENTS

The authors would like to thank the Deanship of Scientific Research at King Saud University represented by the Research Center at CBA for supporting this research financially.

REFERENCES

1. Hoerl AE, Kennard RW. Ridge regression: biased estimation for non-orthogonal problems. *Technometrics.* (1970) 12:55–67. doi: 10.1080/00401706.1970.10488634
2. Hoerl AE, Kennard RW. Ridge regression: applications to non-orthogonal problems. *Technometrics.* (1970) 12:69–82. doi: 10.1080/00401706.1970.10488635
3. Kibria BMG, Lukman AFA. New ridge-type estimator for the linear regression model: simulations and applications. *Scientifica.* (2020) 2020:9758378. doi: 10.1155/2020/9758378
4. Özkale MR, Kaçiranlar S. The restricted, and unrestricted two-parameter estimators. *Commun Stat Theory Methods.* (2007) 36:2707–25. doi: 10.1080/03610920701386877
5. Dawoud I, Kibria BMG. A new biased estimator to combat the multicollinearity of the gaussian linear regression model. *Stat J.* (2020) 3:526–41. doi: 10.3390/stats3040033
6. Roozbeh M, Arashi M, Hamzah NA. Generalized cross-validation for simultaneous optimization of tuning parameters in ridge regression. *Iran J Sci Technol Trans A Sci.* (2020) 44:473–85. doi: 10.1007/s40995-020-00851-1
7. Lukman AF, Ayinde K, Kibria GBM, Adewuyi E. Modified ridge-type estimator for the gamma regression model. *Commun Stat Simul Comput.* (2020). doi: 10.1080/03610918.2020.1752720
8. Arashi M, Roozbeh M, Hamzah NA, Gasparini M. Ridge regression and its applications in genetic studies. *PloS One.* (2021) 16:e0245376. doi: 10.1371/journal.pone.0245376
9. Farghali RA, Qasim M, Kibria BG, Abonazel MR. Generalized two-parameter estimators in the multinomial logit regression model: methods, simulation and application. *Commun Stat Simul Comput.* (2021) 1−16. doi: 10.1080/03610918.2021.1934023
10. Lukman AF, Aladeitan B , Ayinde K, Abonazel MR. Modified ridge-type for the Poisson regression model: simulation and application. *J Appl Stat.* (2021). 1−13. doi: 10.1080/02664763.2021.1889998
11. Lukman AF, Issam D, Kibria GBM, Zakariya A, Aladeitan B. A new ridge-type estimator for the gamma regression model. *Scientifica.* (2021) 2021:1–8. doi: 10.1155/2021/5545356
12. Algamal ZY, Abonazel MR. Developing a Liu-type estimator in beta regression model. *Concurrency Comput Pract Exp.* (2021) 34:e6685. doi: 10.1002/cpe.6685
13. Akram MN, Amin M, Elhassanein A, Ullah MA. A new modified ridge-type estimator for the beta regression model: simulation and application. *AIMS Math.* (2022) 7:1035–57. doi: 10.3934/math.2022062
14. Abonazel MR, Algamal ZY, Awwad FA, Taha IM. A new two-parameter estimator for beta regression model: method, simulation, and application. *Front Appl Math Stat.* (2022) 7:780322. doi: 10.3389/fams.2021.780322
15. Ferrari S, Cribari-Neto F. Beta regression for modelling rates and proportions. *J Appl Stat.* (2004) 31:799–815. doi: 10.1080/0266476042000214501
16. Algamal ZY. A particle swarm optimization method for variable selection in beta regression model. *Electron J Appl Stat Anal.* (2019) 12:508–19.
17. Mahmood SW, Seyala NN, Algamal ZY. Adjusted R2-type measures for beta regression model. *Electron J Appl Stat Anal.* (2020) 13:350–7. doi: 10.1285/i20705948v13n2p350
18. Espinheira PL, Ferrari SL, Cribari-Neto F. On beta regression residuals. *J Appl Stat.* (2008) 35:407–19. doi: 10.1080/02664760701834931
19. Espinheira PL, da Silva LCM, Silva ADO. *Prediction Measures in Beta Regression Models*. arXiv *preprint arXiv*:1501.04830 (2015).
20. Espinheira PL, da Silva LCM, Silva ADO, Ospina R. Model selection criteria on beta regression for machine learning. *Mach Learn Knowl Extraction.* (2019) 1:427–49. doi: 10.3390/make1010026
21. Abonazel MR, Taha IM. Beta ridge regression estimators: simulation and application. *Commun Stat Simul Comput.* (2021) 1−13. doi: 10.1080/03610918.2021.1960373
22. Qasim M, Månsson K, Golam Kibria BM. On some beta ridge regression estimators: method, simulation and application. *J Stat Comput Simul.* (2021) 91:1699–712. doi: 10.1080/00949655.2020.1867549
23. Lukman AF, Adewuyi E, Månsson K, Kibria GBM. A new estimator for the multicollinear Poisson regression model: simulation and application. *Sci Rep.* (2021) 11:3732. doi: 10.1038/s41598-021-82582-w
24. Hoerl AE, Kennard RW, Baldwin KF. Ridge regression: some simulations. *Commun Stat Theory Methods.* (1975) 4:105–23. doi: 10.1080/03610917508548342
25. Lukman AF, Ayinde K, Binuomote S, Onate AC. Modified ridge-type estimator to combat multicollinearity: application to chemical data. *J Chemometr.* (2019) 33:e3125. doi: 10.1002/cem.3125
26. Kibria BG. Performance of some new ridge regression estimators. *Commun Stat Simul Comput.* (2003) 32:419–35. doi: 10.1081/SAC-120017499
27. Kaçiranlar S, Dawoud I. On the performance of the Poisson and the negative binomial ridge predictors. *Commun Stat Simul Comput.* (2018) 47:1751–70. doi: 10.1080/03610918.2017.1324978
28. Rady EA, Abonazel MR, Taha IM. A new biased estimator for zero-inflated count regression models. *J Mod Appl Stat Methods.* (2019). Available online at: https://www.researchgate.net/publication/337155202_A_New_Biased_Estimator_for_Zero-Inflated_Count_Regression_Models
29. Abonazel MR, Farghali RA. Liu-Type multinomial logistic estimator. *Sankhya B.* (2019) 81:203–25. doi: 10.1007/s13571-018-0171-4
30. Dawoud I, Abonazel MR. Robust Dawoud–Kibria estimator for handling multicollinearity and outliers in the linear regression model. *J Stat Comput Simul.* (2021) 91:3678–92. doi: 10.1080/00949655.2021.1945063
31. Awwad FA, Dawoud I, Abonazel MR. Development of robust Özkale–Kaçiranlar and Yang–Chang estimators for regression models in the presence of multicollinearity and outliers. *Concurrency Comput Pract Exp.* (2021) e6779. doi: 10.1002/cpe.6779

32. Akta,ş S, Unlu H. Beta regression for the indicator values of well-being index for provinces in Turkey. *J Eng Technol Appl Sci.* (2017) 2:101–11. doi: 10.30931/jetas.321165
33. Kim JH. Multicollinearity and misleading statistical results. *Korean J Anesthesiol.* (2019) 72:558–69. doi: 10.4097/kja.19087
34. Amini M, Roozbeh M. Optimal partial ridge estimation in restricted semiparametric regression models. *J Multivariate Anal.* (2015) 136:26–40. doi: 10.1016/j.jmva.2015.01.005
35. Roozbeh M. Optimal QR-based estimation in partially linear regression models with correlated errors using GCV criterion. *Comput Stat Data Anal.* (2018) 117:45–61. doi: 10.1016/j.csda.2017.08.002
36. Uslu VR, Egrioglu E, Bas E. Finding optimal value for the shrinkage parameter in ridge regression via particle swarm optimization. *Am J Intell Syst.* (2014) 4:142–7. doi: 10.5923/j.ajis.20140404.03
37. Inan D, Egrioglu E, Sarica B, Askin OE, Tez M. Particle swarm optimization based Liu-type estimator. *Commun Stat Theory Methods.* (2017) 46:11358–69. doi: 10.1080/03610926.2016.1267759

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Applied Mathematics and Statistics

ORIGINAL RESEARCH
published: 17 March 2022
doi: 10.3389/fams.2022.830530

Instantaneous Frequency-Embedded Synchrosqueezing Transform for Signal Separation

Qingtang Jiang[1]*, Ashley Prater-Bennette[2], Bruce W. Suter[3] and Abdelbaset Zeyani[4]

[1] Department of Mathematics and Statistics, University of Missouri–St. Louis, St. Louis, MO, United States, [2] The Air Force Research Laboratory, AFRL/RISB, Rome, NY, United States, [3] The Air Force Research Laboratory, AFRL, Rome, NY, United States, [4] Department of Mathematics, Wichita State University, Wichita, KS, United States

The synchrosqueezing transform (SST) and its variants have been developed recently as an alternative to the empirical mode decomposition scheme to model a non-stationary signal as a superposition of amplitude- and frequency-modulated Fourier-like oscillatory modes. In particular, SST performs very well in estimating instantaneous frequencies (IFs) and separating the components of non-stationary multicomponent signals with slowly changing frequencies. However its performance is not desirable for signals having fast-changing frequencies. Two approaches have been proposed for this issue. One is to use the 2nd-order or high-order SST, and the other is to apply the instantaneous frequency-embedded SST (IFE-SST). For the SST or high order SST approach, one single phase transformation is applied to estimate the IFs of all components of a signal, which may yield not very accurate results in IF estimation and component recovery. IFE-SST uses an estimation of the IF of a targeted component to produce accurate IF estimation. The phase transformation of IFE-SST is associated with the targeted component. Hence the IFE-SST has certain advantages over SST in IF estimation and signal separation. In this article, we provide theoretical study on the instantaneous frequency-embedded short-time Fourier transform (IFE-STFT) and the associated SST, called IFE-FSST. We establish reconstructing properties of IFE-STFT with integrals involving the frequency variable only and provide reconstruction formula for individual components. We also consider the 2nd-order IFE-FSST.

Keywords: short-time Fourier transform, synchrosqueezing transform, instantaneous frequency-embedded STFT, instantaneous frequency-embedded SST, instantaneous frequency estimation

AMS Mathematics Subject Classification: 42C15, 42A38

OPEN ACCESS

Edited by:
Hau-Tieng Wu,
Duke University, United States

Reviewed by:
Duong Hung Pham,
UMR5505 Institut de Recherche en Informatique de Toulouse (IRIT), France
Anouar Ben Mabrouk,
University of Kairouan, Tunisia

****Correspondence:***
Qingtang Jiang
jiangq@umsl.edu

Specialty section:
This article was submitted to Mathematics of Computation and Data Science, a section of the journal Frontiers in Applied Mathematics and Statistics

Received: *07 December 2021*
Accepted: *17 February 2022*
Published: *17 March 2022*

Citation:
Jiang Q, Prater-Bennette A, Suter BW and Zeyani A (2022) Instantaneous Frequency-Embedded Synchrosqueezing Transform for Signal Separation. Front. Appl. Math. Stat. 8:830530. doi: 10.3389/fams.2022.830530

1. INTRODUCTION

Recently the continuous wavelet transform-based synchrosqueezed transform (WSST) was developed in [1] as an empirical mode decomposition (EMD)-like tool to model a non-stationary signal $x(t)$ as

$$x(t) = A_0(t) + \sum_{k=1}^{K} x_k(t), \qquad x_k(t) = A_k(t)e^{i2\pi\phi_k(t)}, \tag{1}$$

with $A_k(t), \phi_k'(t) > 0$, where $A_k(t)$ is called the instantaneous amplitudes and $\phi_k'(t)$ the instantaneous frequencies (IFs). The representation (1) of non-stationary signals is important to extract information hidden in $x(t)$. WSST not only sharpens the time-frequency representation of a signal, but also recovers the components of a multicomponent signal. The synchrosqueezing transform (SST) provides an alternative to the EMD method introduced in [2] and its variants considered in many articles such as [3–12], and it overcomes some limitations of the EMD and ensemble EMD schemes such as mode-mixing. Many works on SST have been carried out since the publication of the seminal article [1]. For example, the short-time Fourier transform (STFT)-based SST (FSST) [13–15], the 2nd-order SST [16–18], the higher-order FSST [19, 20], a hybrid EMD-WSST [21], the WSST with vanishing moment wavelets [22], the multitapered SST [23], the synchrosqueezed wave packet transform [24] and the synchrosqueezed curvelet transform [25] were proposed. Furthermore, the adaptive SST with a window function having a changing parameter was proposed in [26–31]. SST has been successfully used in machine fault diagnosis [32, 33], and medical data analysis applications [see [34] and references therein]. [35] proposed a direct time-frequency method (called SSO) based on the ridges of spectrogram for signal separation. This method has been extended recently to the linear chirp-based models [36, 37] and the models based on the CWT scaleogram [38, 39]. A hybrid EMD-SSO computational scheme was developed in [40].

If the IFs $\phi_k'(t)$ of the components $x_k(t)$ of a non-stationary multicomponent signal change slowly or change slowly compared with $\phi_k(t)$, then SST performs very well in estimating $\phi_k'(t)$ and separating the components $x_k(t)$ from $x(t)$. However its performance is not desirable for signals having fast-changing frequencies. The 2nd-order and high-order SSTs were proposed for this issue and they do improve the accuracy of IF estimation and component recovery. The problem with the 2nd-order and high-order SSTs is that, like the convectional SST, one single phase transformation is applied to estimate the IFs of all components of a signal, which may not yield desirable results in IF estimation or component recovery.

Another approach is to demodulate the original signal to change a wide-band component into a narrow-band component. Li and Liang [41] and Meignen et al. [42] demodulate the original signal into a pure carrier signal and apply WSST and the 2nd-order FSST to the demodulated signal, respectively. FSST based on another demodulation was proposed in [43]. The demodulation introduced in [43] transforms a one-dimensional signal, as a function of time only, into a two-dimensional bivariate function of time and time-shift. The STFT of the demodulated signal has more concentrated time-frequency representation than the conventional STFT, and in the meantime it well characterizes time-frequency properties of the signal [43]. The demodulation approach of [43] is considered in [44] in the setting of CWT. The associated CWT and SST are called in [44] the instantaneous frequency-embedded CWT (IFE-CWT) and IFE-SST, respectively. For consistency, we call the STFT of the demodulated signal and the associated FSST in [43]: the IFE-STFT and IFE-FSST respectively. [43] shows that IFE-FSST results in sharp time-frequency representations of signals. However component recovery of a multicomponent signal was not discussed in [43]. In this article, we consider theoretical analysis of IFE-STFT for establishing the component recovery with IFE-FSST. Compared with the study of IFE-SST in [44], we derive in this article mathematically rigorous phase transformation for IFE-FSST. In addition, in this article we also consider the 2nd-order IFE-FSST and derive the associate phase transformation.

The rest of this article is organized as follows. In Section 2, we briefly review FSST and the 2nd-order FSST. After that, we consider in Section 3 the IFE-STFT and establish reconstructing properties of IFE-STFT with integrals involving the frequency variable only. In Section 4, we derive mathematically rigorous phase transformations for IFE-FSST and the 2nd-order IFE-FSST. In addition, we provide reconstruction formula for individual components. Implementations and IFE-FSST-based component recovery algorithms are discussed in Section 5. Some experimental results are also provided in Section 5.

2. SHORT-TIME FOURIER TRANSFORM-BASED SST

The (modified) short-time Fourier transform (STFT) of $x(t)$ is defined by

$$V_x(t,\eta) := \int_{-\infty}^{\infty} x(\tau) g(\tau - t) e^{-i2\pi\eta(\tau-t)} d\tau, \tag{2}$$

where $g(t)$ is a window function with $g(0) \neq 0$. $x(t)$ can be reconstructed from its STFT:

$$x(t) = \frac{1}{\|g\|_2^2} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} V_x(t,\xi)\overline{g(t-\tau)} e^{-i2\pi\xi(\tau-t)} d\tau\, d\xi. \tag{3}$$

$x(t)$ can also be recovered back from its STFT with an integral involving only the frequency variable η:

$$x(t) = \frac{1}{g(0)} \int_{-\infty}^{\infty} V_x(t,\eta) d\eta. \tag{4}$$

In addition, one can show that if $g(t)$ and $x(t)$ are real-valued, then

$$x(t) = \frac{2}{g(0)} \mathrm{Re}\Big(\int_{0}^{\infty} V_x(t,\eta) d\eta \Big). \tag{5}$$

Furthermore, one can verify that STFT can be written as

$$V_x(t,\eta) = \int_{-\infty}^{\infty} \widehat{x}(\xi)\widehat{g}(\eta-\xi) e^{i2\pi t\xi} d\xi. \tag{6}$$

The STFT $V_x(t,\eta)$ of a slowly growing $x(t)$ is well-defined and the above formulas still hold if the window function $g(t)$ has certain smoothness and certain decaying order as $t \to \infty$, for example $g(t)$ is in the Schwarz class $\mathcal{S}$. In this article, unless otherwise stated, we always assume that a window function $g(t)$ has certain smoothness and decaying properties and $g(0) \neq 0$, and assume that a signal $x(t)$ is a slowly growing function.

2.1. FSST

The idea of FSST is to re-assign the frequency variable η of $V_x(t,\eta)$. First we look at the STFT of $x(t) = Ae^{i2\pi\xi_0 t}$, where ξ_0 is a positive constant. With

$$\begin{aligned} V_x(t,\eta) &= \int_{-\infty}^{\infty} Ae^{i2\pi\xi_0\tau} g(\tau - t)e^{-i2\pi\eta(\tau-t)}d\tau \\ &= A\widehat{g}(\eta - \xi_0)e^{i2\pi t\xi_0}, \end{aligned}$$

we can obtain the IF ξ_0 of $x(t)$ by

$$\frac{\partial_t V_x(t,\eta)}{2\pi i V_x(t,\eta)} = \xi_0,$$

where throughout this article, ∂_t denotes the partial derivative with respect to variable t. For a general $x(t)$, at (t,η) for which $V_x(t,\eta) \neq 0$, a good candidate for the IF of $x(t)$ is

$$\frac{\partial_t V_x(t,\eta)}{2\pi i V_x(t,\eta)}.$$

In the following, denote

$$\omega_x(t,\eta) := \mathrm{Re}\Big\{\frac{\partial_t V_x(t,\eta)}{2\pi i V_x(t,\eta)}\Big\}, \quad \text{for } (t,\eta) \text{ with } V_x(t,\eta) \neq 0,$$

which is called the "phase transformation" [1], "instantaneous frequency information" [13], or the "reference IF function" in [21]. FSST is to re-assign the frequency variable η by transforming the STFT $V_x(t,\eta)$ of $x(t)$ to a quantity, denoted by $R_x^{\lambda,\gamma}(t,\xi)$, on the time-frequency plane defined by

$$R_x^{\lambda,\gamma}(t,\xi) := \int_{\{\eta : |V_x(t,\eta)| > \gamma\}} V_x(t,\eta)\frac{1}{\lambda}h\left(\frac{\xi - \omega_x(t,\eta)}{\lambda}\right)d\eta,$$

where ξ is the frequency variable, $h(t)$ a compactly supported function with certain smoothness and $\int_{-\infty}^{\infty} h(t)dt = 1$, $\gamma > 0$ is the threshold for zero and $\lambda > 0$ is a dilation. As $\lambda, \gamma \to 0$, FSST is rewritten as

$$R_x(t,\xi) := \int_{\{\eta : V_x(t,\eta) \neq 0\}} V_x(t,\eta)\delta\big(\omega_x(t,\eta) - \xi\big)d\eta. \tag{7}$$

For simplicity of presentation, throughout this article SSTs will be expressed as (7).

Due to (4), we have that the input signal $x(t)$ can be recovered from its FSST by

$$x(t) = \frac{1}{g(0)}\int_{-\infty}^{\infty} R_x(t,\xi)d\xi. \tag{8}$$

If in addition, $g(t)$ and $x(t)$ are real-valued, then by (5),

$$x(t) = \frac{2}{g(0)}\mathrm{Re}\Big(\int_0^{\infty} R_x(t,\xi)d\xi\Big). \tag{9}$$

For a multicomponent signal $x(t)$ given by (1), when $A_k(t), \phi_k(t)$ satisfy certain conditions, each component $x_k(t)$ can be recovered from its FSST:

$$x_k(t) \approx \frac{1}{g(0)}\int_{|\xi - \mathrm{IF}_k(t)| < \Gamma} R_x(t,\xi)d\xi, \tag{10}$$

for certain $\Gamma > 0$, where $\mathrm{IF}_k(t)$ is an estimate to $\phi_k'(t)$. See [13–15] for the details.

In practice, t, η, ξ are discretized. Suppose t_n, η_j, ξ_m are the sampling points of t, η, ξ respectively. Then the FSST of $x(t)$ is given by

$$R_x(t_n,\xi_m) = \sum_{j:\ |\omega_x(t_n,\eta_j) - \xi_m| \leq \Delta\xi/2, |V_x(t_n,\eta_j)| \geq \gamma} V_x(t_n,\eta_j)\triangle\eta_j,$$

where $\triangle\eta_j = \eta_j - \eta_{j-1}$, and $\gamma > 0$ is a threshold for the condition $|V_x(t,\eta)| > 0$. The recovering formulas (8) and (9) result in

$$x(t_n) = \frac{1}{g(0)}\sum_m R_x(t_n,\xi_m)\triangle\xi_m,$$

and for real-valued $g(t)$ and $x(t)$,

$$x(t_n) = \frac{2}{g(0)}\mathrm{Re}\Big(\sum_m R_x(t_n,\xi_m)\triangle\xi_m\Big),$$

where $\triangle\xi_m = \xi_m - \xi_{m-1}$.

2.2. Second-Order FSST

The 2nd-order FSST was introduced in [16]. The main idea is to define a new phase transformation $\omega_x^{2\mathrm{nd}}$ such that when $x(t)$ is a linear frequency modulation (LFM) signal (also called a linear chirp), then $\omega_x^{2\mathrm{nd}}$ is exactly the IF of $x(t)$. We say $x(t)$ is a LFM signal or a linear chirp if

$$x(t) = Ae^{i2\pi\phi(t)} = Ae^{i2\pi(ct + \frac{1}{2}rt^2)} \tag{11}$$

with phase function $\phi(t) = ct + \frac{1}{2}rt^2$, IF $\phi'(t) = c + rt$ and chirp rate $\phi''(t) = r$. In [16], the reassignment operators are used to derive $\omega_x^{2\mathrm{nd}}$. Different phase transformation $\omega_x^{2\mathrm{nd}}$ for the 2nd-order SST can be derived without using the reassignment operators see [28, 29].

Let g be a given window function. Denote

$$g_1(t) = tg(t). \tag{12}$$

Recall that $V_x(t,\eta)$ denotes the STFT of $x(t)$ with g defined by (2). In this article, we let $V_x^{g_1}(t,\eta)$ denote the STFT of $x(t)$ with $g_1(t)$, namely, the integral on the right-hand side of (2) with $g(t)$ replaced by $g_1(t)$. Define

$$\omega_x^{2\mathrm{nd}}(t,\eta) := \begin{cases} \mathrm{Re}\Big\{\frac{\partial_t V_x(t,\eta)}{i2\pi V_x(t,\eta)}\Big\} - \mathrm{Re}\Big\{q_0(t,\eta)\frac{V_x^{g_1}(t,\eta)}{i2\pi V_x(t,\eta)}\Big\}, \\ \quad \text{if } \partial_\eta\big(\frac{V_x^{g_1}(t,\eta)}{V_x(t,\eta)}\big) \neq 0, V_x(t,\eta) \neq 0, \\ \mathrm{Re}\Big\{\frac{\partial_t V_x(t,\eta)}{i2\pi V_x(t,\eta)}\Big\}, \\ \quad \text{if } \partial_\eta\big(\frac{V_x^{g_1}(t,\eta)}{V_x(t,\eta)}\big) = 0, V_x(t,\eta) \neq 0, \end{cases} \tag{13}$$

where

$$q_0(t,\eta) := \frac{1}{\partial_\eta\big(\frac{V_x^{g_1}(t,\eta)}{V_x(t,\eta)}\big)}\partial_\eta\Big(\frac{\partial_t V_x(t,\eta)}{V_x(t,\eta)}\Big).$$

Then one can show that $\omega_x^{2\mathrm{nd}}(t,\eta)$ is exactly the IF $\phi'(t)$ of $x(t)$ if $x(t)$ is an LFM signal given by (11), see [19, 28]. Thus, we may define $\omega_x^{2\text{nd}}(t,\eta)$ in (13) as the phase transformation for the 2nd-order FSST. Very recently a simple phase transformation for the 2nd-order FSST was proposed in [18].

3. INSTANTANEOUS FREQUENCY-EMBEDDED STFT

IFE-FSST is based on the IFE-STFT, which is defined below.

Definition 1. *Suppose $\varphi(t)$ is a differentiable function with $\varphi'(t) > 0$. Let $\eta_0 > 0$. The IFE-STFT of $x(t) \in L_2(\mathbb{R})$ with $\varphi(t), \eta_0$ and a window function $g(t)$ is defined by*

$$V_x^{\mathrm{I}}(t,\eta) := \int_{-\infty}^{\infty} x(\tau)e^{-i2\pi\left(\varphi(\tau)-\varphi(t)-\varphi'(t)(\tau-t)-\eta_0\tau\right)} g(\tau-t)e^{-i2\pi\eta(\tau-t)}d\tau. \qquad (14)$$

In the above definition, we assume $x(t) \in L_2(\mathbb{R})$. The definition of IFE-STFT can be extended to slowly growing functions $x(t)$ if $g(t)$ has certain smoothness and certain decaying order as $t \to \infty$.

Li and Liang [41] proposed the modulation $x(\tau) \to \widetilde{x}(\tau) = x(\tau)e^{-i2\pi(\varphi(\tau)-\eta_0\tau)}$ and applied WSST to the modulated signal $\widetilde{x}(\tau)$, while [42] applied the 2nd-order FSST to $\widetilde{x}(t)$. The modulation:

$$x(\tau) \to x(\tau)e^{-i2\pi\left(\varphi(\tau)-\varphi(t)-\varphi'(t)(\tau-t)-\eta_0\tau\right)}$$

introduced in [43] for IFE-FSST and also used in [44] for IFE-WSST is different from that used in [41, 42]. IFE-STFT and IFE-CWT with such a modulation not only have more concentrated time-frequency representation than the conventional STFT and CWT respectively, but also well keep the IF of the signal. The reader is referred to [43] and [44] for detailed discussions.

[43] provides a reconstruction formula with IFE-STFT for the whole signal $x(t)$, which is similar to (3) and involves an integral with both the time and frequency variables. [43] does not consider individual component recovery formula with IFE-FSST. In this article, we provide such a component recovery formula. To this regard, in this section we establish a reconstruction formula with IFE-STFT like (4), which involves an integral with the frequency variable only. First we have the following property about the IFE-STFT.

Proposition 1. *Let $V_x^{\mathrm{I}}(t,\eta)$ be the IFE-STFT of $x(t)$ defined by (14). Then*

$$V_x^{\mathrm{I}}(t,\eta) = e^{i2\pi\varphi(t)}\int_{-\infty}^{\infty}\widehat{\widetilde{x}}(\xi)\widehat{g}(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}d\xi, \qquad (15)$$

where

$$\widetilde{x}(t) = x(t)e^{-i2\pi(\varphi(t)-\eta_0 t)}. \qquad (16)$$

Proof: We have

$$\begin{aligned}
V_x^{\mathrm{I}}(t,\eta) &= e^{i2\pi\varphi(t)}\int_{-\infty}^{\infty}\widetilde{x}(\tau)e^{i2\pi\varphi'(t)(\tau-t)}g(\tau-t)e^{-i2\pi\eta(\tau-t)}d\tau \\
&= e^{i2\pi\varphi(t)}\int_{-\infty}^{\infty}\widetilde{x}(\tau)g(\tau-t)e^{-i2\pi\left(\eta-\varphi'(t)\right)(\tau-t)}d\tau \\
&= e^{i2\pi\varphi(t)}\int_{-\infty}^{\infty}\widehat{\widetilde{x}}(\xi)\widehat{g}\big(\eta-\varphi'(t)-\xi\big)e^{i2\pi t\xi}d\xi,
\end{aligned}$$

where the last equality follows from (6). □

The next theorem shows that $x(t)$ can be recovered from its IFE-STFT with an integral involving η only.

Theorem 1. *Let $x(t)$ be a function in $L_2(\mathbb{R})$. Then*

$$x(t) = \frac{e^{-i2\pi\eta_0 t}}{g(0)}\int_{-\infty}^{\infty}V_x^{\mathrm{I}}(t,\eta)d\eta. \qquad (17)$$

Proof: Let $\widetilde{x}(t)$ be the function defined by (16). From (15), we have

$$\begin{aligned}
\int_{-\infty}^{\infty}V_x^{\mathrm{I}}(t,\eta)d\eta &= e^{i2\pi\varphi(t)}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\widehat{\widetilde{x}}(\xi)\widehat{g}(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}d\xi d\eta \\
&= e^{i2\pi\varphi(t)}\int_{-\infty}^{\infty}\widehat{\widetilde{x}}(\xi)\int_{-\infty}^{\infty}\widehat{g}(\eta-\varphi'(t)-\xi)d\eta\, e^{i2\pi t\xi}d\xi \\
&= e^{i2\pi\varphi(t)}g(0)\int_{-\infty}^{\infty}\widehat{\widetilde{x}}(\xi)e^{i2\pi t\xi}d\xi \\
&= e^{i2\pi\varphi(t)}g(0)\widetilde{x}(t) \\
&= e^{i2\pi\varphi(t)}g(0)x(t)e^{-i2\pi\left(\varphi(t)-\eta_0 t\right)} \\
&= g(0)x(t)e^{i2\pi\eta_0 t}.
\end{aligned}$$

Thus, Equation (17) holds. □

If one is interested in $V_x^{\mathrm{I}}(t,\eta)$ with the positive frequency $\eta > 0$ only, then we have the following result on how to recover $x(t)$ from $V_x^{\mathrm{I}}(t,\eta)$.

Theorem 2. *Suppose $supp(\widehat{g}) \subseteq [-\Delta,\Delta]$ for some Δ, and $\varphi'(t) \geq \Delta$. Let $y(t) = x(t)e^{-i2\pi\varphi(t)}$. If $\widehat{y}(\eta) = 0$, $\eta \leq B$ for some constant B, then for any $\eta_0 \geq -B$,*

$$x(t) = \frac{e^{-i2\pi\eta_0 t}}{g(0)}\int_{0}^{\infty}V_x^{\mathrm{I}}(t,\eta)d\eta. \qquad (18)$$

Proof: Let $\widetilde{x}(t)$ be the function defined by (16). Then $\widetilde{x}(t) = y(t)e^{i2\pi\eta_0 t}$. Thus, $\widehat{\widetilde{x}}(\xi) = \widehat{y}(\xi-\eta_0)$. Therefore, from (15), we have

$$\begin{aligned}
\int_{0}^{\infty}V_x^{\mathrm{I}}(t,\eta)d\eta &= e^{i2\pi\varphi(t)}\int_{0}^{\infty}\int_{-\infty}^{\infty}\widehat{\widetilde{x}}(\xi)\widehat{g}(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}d\xi d\eta \\
&= e^{i2\pi\varphi(t)}\int_{0}^{\infty}\int_{-\infty}^{\infty}\widehat{y}(\xi-\eta_0)\widehat{g}(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}d\xi d\eta \\
&= e^{i2\pi\varphi(t)}\int_{0}^{\infty}\int_{-\infty}^{\infty}\widehat{y}(\xi)\widehat{g}(\eta-\varphi'(t)-\xi-\eta_0)e^{i2\pi t(\xi+\eta_0)}d\xi d\eta \\
&= e^{i2\pi\left(\varphi(t)+t\eta_0\right)}\int_{-\infty}^{\infty}\widehat{y}(\xi)\int_{0}^{\infty}\widehat{g}(\eta-\varphi'(t)-\xi-\eta_0)e^{i2\pi t\xi}d\eta d\xi \\
&= e^{i2\pi\left(\varphi(t)+t\eta_0\right)}\int_{B}^{\infty}\widehat{y}(\xi)e^{i2\pi t\xi}\int_{0}^{\infty}\widehat{g}(\eta-\varphi'(t)-\xi-\eta_0)d\eta d\xi.
\end{aligned}$$

When $\xi \geq B$ and $\eta_0 \geq -B$, we have $-\varphi'(t) - \xi - \eta_0 \leq -\Delta - B + B = -\Delta$. This and the assumption $\mathrm{supp}(\widehat{g}) \subseteq [-\Delta, \Delta]$ lead to

$$\begin{aligned}\int_0^\infty \widehat{g}(\eta - \varphi'(t) - \xi - \eta_0)d\eta &= \int_{-\varphi'(t)-\xi-\eta_0}^\infty \widehat{g}(\eta)d\eta \\ &= \int_{-\infty}^\infty \widehat{g}(\eta)d\eta = g(0).\end{aligned}$$

Hence,

$$\begin{aligned}\int_0^\infty V_x^{\mathrm{I}}(t,\eta)d\eta &= e^{i2\pi\left(\varphi(t)+t\eta_0\right)} \int_B^\infty \widehat{y}(\xi)e^{i2\pi t\xi} g(0)d\xi \\ &= e^{i2\pi\left(\varphi(t)+t\eta_0\right)} g(0) \int_{-\infty}^\infty \widehat{y}(\xi)e^{i2\pi t\xi} d\xi \\ &= e^{i2\pi\left(\varphi(t)+t\eta_0\right)} g(0)y(t) \\ &= e^{i2\pi\left(\varphi(t)+t\eta_0\right)} g(0)x(t)e^{-i2\pi\varphi(t)} \\ &= g(0)x(t)e^{i2\pi\eta_0 t}.\end{aligned}$$

Thus, Equation (18) holds. □

Next theorem shows that when the condition $\widehat{y}(\eta) = 0, \eta \leq B$ in Theorem 2 does not hold, the integral in the right-hand side of (18) can still approximate $x(t)$ well if η_0 is large.

Theorem 3. *Let $y(t) = x(t)e^{-i2\pi\varphi(t)}$. Then*

$$x(t) = \frac{e^{-i2\pi\eta_0 t}}{g(0)} \int_0^\infty V_x^{\mathrm{I}}(t,\eta)d\eta + \mathrm{Err}, \qquad (19)$$

with

$$|\mathrm{Err}| \leq \frac{\int_{-\infty}^\infty |\widehat{g}(\xi)|d\xi}{g(0)} \int_{-\infty}^{-\eta_0} |\widehat{y}(\xi)|d\xi.$$

Proof: By Theorem 1,

$$\begin{aligned}\int_0^\infty V_x^{\mathrm{I}}(t,\eta)d\eta &= \int_{-\infty}^\infty V_x^{\mathrm{I}}(t,\eta)d\eta - \int_{-\infty}^0 V_x^{\mathrm{I}}(t,\eta)d\eta \\ &= e^{i2\pi\eta_0 t}g(0)x(t) - \int_{-\infty}^0 V_x^{\mathrm{I}}(t,\eta)d\eta.\end{aligned}$$

Thus,

$$\mathrm{Err} = \frac{e^{-i2\pi\eta_0 t}}{g(0)} \int_{-\infty}^0 V_x^{\mathrm{I}}(t,\eta)d\eta.$$

With

$$\begin{aligned}\left|\int_{-\infty}^0 V_x^{\mathrm{I}}(t,\eta)d\eta\right| &= \left|e^{i2\pi\varphi(t)} \int_{-\infty}^0 \int_{-\infty}^\infty \widehat{y}(\xi - \eta_0)\widehat{g}(\eta - \varphi'(t) - \xi)e^{i2\pi t\xi} d\xi d\eta\right| \\ &\leq \int_{-\infty}^0 \int_{-\infty}^\infty |\widehat{y}(\xi - \eta_0)|\, |\widehat{g}(\eta - \varphi'(t) - \xi)e^{i2\pi t\xi}|d\eta d\xi \\ &\leq \int_{-\infty}^0 |\widehat{y}(\xi - \eta_0)| \int_{-\infty}^\infty |\widehat{g}(\eta - \varphi'(t) - \xi)|d\eta d\xi \\ &= \int_{-\infty}^\infty |\widehat{g}(\eta)|d\eta \int_{-\infty}^0 |\widehat{y}(\xi - \eta_0)|d\xi \\ &= \int_{-\infty}^\infty |\widehat{g}(\eta)|d\eta \int_{-\infty}^{-\eta_0} |\widehat{y}(\xi)|d\xi,\end{aligned}$$

we conclude that (19) holds. □

4. IFE-STFT BASED SYNCHROSQUEEZING TRANSFORM

In this section, we consider IFE-FSST, the synchrosqueezing transform based on IFE-STFT. First we show how to derive the phase transformation associated with (the 1st-order) IFE-FSST. After that we introduce the 2nd-order IFE-FSST.

4.1. IFE-FSST

To define IFE-FSST, first we need to define the corresponding phase transformation $\omega_x^{\mathrm{I}}(a,b)$. Let us consider the case $x(t) = Ae^{i2\pi\xi_0 t}$ for some $\xi_0 > 0$. With $x'(t) = i2\pi\xi_0\, x(t)$, we have

$$V_{x'}^{\mathrm{I}}(t,\eta) = i2\pi\xi_0 V_x^{\mathrm{I}}(t,\eta).$$

On the other hand,

$$\begin{aligned}V_{x'}^{\mathrm{I}}(t,\eta) &= \int_{-\infty}^\infty \partial_\tau\left(x(t+\tau)\right)e^{-i2\pi\left(\varphi(t+\tau)-\varphi(t)-\varphi'(t)\tau-\eta_0(t+\tau)\right)} g(\tau)e^{-i2\pi\eta\tau} d\tau \\ &= -\int_{-\infty}^\infty x(t+\tau)\partial_\tau\left(e^{-i2\pi\left(\varphi(t+\tau)-\varphi(t)-\varphi'(t)\tau-\eta_0(t+\tau)\right)} g(\tau)e^{-i2\pi\eta\tau}\right)d\tau \\ &= -\int_{-\infty}^\infty x(t+\tau)(-i2\pi)\left(\varphi'(t+\tau) - \varphi'(t) - \eta_0 + \eta\right) \\ &\quad e^{-i2\pi\left(\varphi(t+\tau)-\varphi(t)-\varphi'(t)\tau-\eta_0(t+\tau)+\eta)\right)} g(\tau)d\tau \\ &\quad -\int_{-\infty}^\infty x(t+\tau)e^{-i2\pi\left(\varphi(t+\tau)-\varphi(t)-\varphi'(t)\tau-\eta_0(t+\tau)+\eta)\right)} g'(\tau)d\tau \\ &= i2\pi V_{x\varphi'}^{\mathrm{I}}(t,\eta) + i2\pi(\eta - \varphi'(t) - \eta_0)V_x^{\mathrm{I}}(t,\eta) - V_x^{I,g'}(t,\eta),\end{aligned} \qquad (20)$$

where $V_x^{\mathrm{I},g'}(t,\eta)$ denotes the IFE-STFT of $x(t)$ defined by (14) with $\varphi(t)$ and the window function g' given by (12). Thus, if $V_x^{\mathrm{I}}(t,\eta) \neq 0$, then

$$\xi_0 = \frac{V_{x'}^{\mathrm{I}}(t,\eta)}{i2\pi V_x^{\mathrm{I}}(t,\eta)} = \frac{i2\pi V_{x\varphi'}^{\mathrm{I}}(t,\eta) - V_x^{\mathrm{I},g'}(t,\eta)}{i2\pi V_x^{\mathrm{I}}(t,\eta)} + \eta - \varphi'(t) - \eta_0.$$

Based on the above discussion, for a general signal $x(t)$, we define the phase transformation $\omega_x^{\mathrm{I}}(a,b)$ of the IFE-FSST of $x(t)$ to be

$$\omega_x^{\mathrm{I}}(t,\eta) := \mathrm{Re}\Big(\frac{i2\pi V_{x\varphi'}^{\mathrm{I}}(t,\eta) - V_x^{\mathrm{I},g'}(t,\eta)}{i2\pi V_x^{\mathrm{I}}(t,\eta)}\Big) + \eta - \varphi'(t) - \eta_0. \qquad (21)$$

Definition 2. *Suppose $\varphi(t)$ is a differentiable function with $\varphi'(t) > 0$. The IFE-FSST of a signal $x(t)$ with φ and ξ_0 is defined by*

$$R_x^{\mathrm{I}}(t,\xi) := \int_{\{\eta\,:\, V_x^{\mathrm{I}}(t,\eta)\neq 0\}} V_x^{\mathrm{I}}(t,\eta)\delta\left(\omega_x^{\mathrm{I}}(t,\eta) - \xi\right)d\eta$$

where $\omega_x^{\mathrm{I}}(t,\eta)$ is the phase transformation defined by (21).

The IFE-FSST is called the demodulation transform-based SST in [43]. The corresponding phase transformation in [43] is different from our $\omega_x^{\mathrm{I}}(t,\eta)$ defined in (21).

By (18) in Theorem 1, we know the input signal $x(t)$ can be recovered from its IFE-FSST as shown in the following:
For $x(t) \in L_2(\mathbb{R})$,

$$x(t) = \frac{e^{-i2\pi\eta_0 t}}{g(0)} \int_{-\infty}^{\infty} R_x^{\mathrm{I}}(t,\xi)d\xi; \tag{22}$$

and if, in addition, the conditions in Theorem 2 hold, then

$$x(t) = \frac{e^{-i2\pi\eta_0 t}}{g(0)} \int_{0}^{\infty} R_x^{\mathrm{I}}(t,\xi)d\xi. \tag{23}$$

For a multicomponent signal $x(t)$ in the form (1), if $R_{x_k}^{\mathrm{I}}(t,\xi)$, $1 \le k \le K$ lie in different time-frequency zones, then following (18), we know $x_k(t)$ can be recovered from its IFE-FSST:

$$x_k(t) \approx \frac{e^{-i2\pi\eta_0 t}}{g(0)} \int_{|\xi-\mathrm{IF}_k(t)|<\Gamma_1} R_x^{\mathrm{I}}(t,\xi)d\xi, \tag{24}$$

for certain $\Gamma_1 > 0$, where $\mathrm{IF}_k(t)$ is an estimate of $\phi_k'(t)$. If $x_k(t)$ and $g(t)$ are real-valued, then

$$x_k(t) \approx \frac{2}{g(0)} \mathrm{Re}\Big(e^{-i2\pi\eta_0 t} \int_{|\xi-\mathrm{IF}_k(t)|<\Gamma_1} R_x^{\mathrm{I}}(t,\xi)d\xi\Big). \tag{25}$$

4.2. 2nd-Order IFE-FSST

In this subsection, we propose the 2nd-order IFE-FSST. The key point is, based on IFE-STFT, to define a phase transformation $\omega_x^{\mathrm{I,2nd}}(t,\eta)$ which is the IF $\phi'(t)$ of $x(t)$ when $x(t)$ is a linear chirp given by (11). As above, for $g_1(t) = tg(t)$, we use $V_x^{\mathrm{I},g_1}(t,\eta)$ to denote the IFE-STFT of $x(t)$ with the window function $g_1(t)$, namely, the integral on the right-hand side of (14) with $g(t)$ replaced by $g_1(t)$. Next we define the phase transformation $\omega_x^{\mathrm{I,2nd}}(t,\eta)$ for the 2nd-order IFE-FSST to be:

$$\omega_x^{\mathrm{I,2nd}}(t,\eta) := \begin{cases} \omega_x^{\mathrm{I}}(t,\eta) - \mathrm{Re}\Big\{Q_0(t,\eta)\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{i2\pi V_x^{\mathrm{I}}(t,\eta)}\Big\}, \\ \qquad \text{if } \partial_\eta\big(\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V_x^{\mathrm{I}}(t,\eta)}\big) \neq 0,\ V_x^{\mathrm{I}}(t,\eta) \neq 0; \\ \\ \omega_x^{\mathrm{I}}(t,\eta), \\ \qquad \text{if } \partial_\eta\big(\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V_x^{\mathrm{I}}(t,\eta)}\big) = 0,\ V_x^{\mathrm{I}}(t,\eta) \neq 0, \end{cases} \tag{26}$$

where $\omega_x^{\mathrm{I}}(t,\eta)$ is defined by (21), and

$$Q_0(t,\eta) := \frac{1}{\partial_\eta\Big(\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V_x^{\mathrm{I}}(t,\eta)}\Big)} \Big\{1 + \partial_\eta\Big(\frac{i2\pi V_{x\varphi'}^{\mathrm{I}}(t,\eta) - V_x^{\mathrm{I},g'}(t,\eta)}{i2\pi V_x^{\mathrm{I}}(t,\eta)}\Big)\Big\}. \tag{27}$$

Theorem 4. *If $x(t)$ is a linear chirp signal given by (11), then at (t,η) where $V_x^{\mathrm{I}}(t,\eta) \neq 0$, $\partial_\eta\big(V_x^{\mathrm{I},g_1}(t,\eta)/V_x^{\mathrm{I}}(t,\eta)\big) \neq 0$, $\omega_x^{\mathrm{I,2nd}}(t,\eta)$ defined by (26) is the IF of $x(t)$, namely $\omega_x^{\mathrm{I,2nd}}(t,\eta) = c + rt$.*

Proof: Here, we provide the proof of $\omega_x^{\mathrm{I,2nd}}(t,\eta) = c+rt$ for more general linear chirp signals given by

$$x(t) = A(t)e^{i2\pi\phi(t)} = Ae^{pt+\frac{q}{2}t^2}e^{i2\pi(ct+\frac{1}{2}rt^2)} \tag{28}$$

where p, q are real numbers.

For the simplicity of presentation, we denote

$$M_{\varphi,g}(\tau,t,\eta) := e^{-i2\pi\big(\varphi(t+\tau)-\varphi(t)-\varphi'(t)\tau-\eta_0(t+\tau)\big)}g(\tau)e^{-i2\pi\eta\tau},$$

and thus, $V_x^{\mathrm{I}}(t,\eta)$ can simply be written as

$$V_x^{\mathrm{I}}(t,\eta) = \int_{-\infty}^{\infty} x(t+\tau)M_{\varphi,g}(\tau,t,\eta)d\tau.$$

Observe that for $x(t)$ given by (28), we have

$$x'(t) = \big(p + qt + i2\pi(c+rt)\big)x(t).$$

Thus,

$$\begin{aligned} V_{x'}^{\mathrm{I}}(t,\eta) &= \int_{-\infty}^{\infty} x'(t+\tau)\, M_{\varphi,g}(\tau,t,\eta)d\tau \\ &= \int_{-\infty}^{\infty} \big(p + q(t+\tau) + i2\pi(c+rt+r\tau)\big) \\ &\qquad x(t+\tau)\, M_{\varphi,g}(\tau,t,\eta)d\tau \\ &= \big(p + qt + i2\pi(c+rt)\big)V_x^{\mathrm{I}}(t,\eta) \\ &\quad +(q+i2\pi r)\int_{-\infty}^{\infty} x(t+\tau)\ \tau M_{\varphi,g}(\tau,t,\eta)\tau d\tau \\ &= \big(p + qt + i2\pi(c+rt)\big)V_x^{\mathrm{I}}(t,\eta) \\ &\quad +(q+i2\pi r)V_x^{\mathrm{I},g_1}(t,\eta). \end{aligned}$$

On the other hand, as shown above, $V_{x'}^{\mathrm{I}}(t,\eta)$ is equal to the quantity in (20). Therefore,

$$\begin{aligned} &\big(p + qt + i2\pi(c+rt)\big)V_x^{\mathrm{I}}(t,\eta) + (q+i2\pi r)V_x^{\mathrm{I},g_1}(t,\eta) \\ &\quad = i2\pi V_{x\varphi'}^{\mathrm{I}}(t,\eta) + i2\pi(\eta - \varphi'(t) - \eta_0)V_x^{\mathrm{I}}(t,\eta) \\ &\quad -V_x^{I,g'}(t,\eta). \end{aligned}$$

Hence, at (t,η) on which $V_x^{\mathrm{I}}(t,\eta) \neq 0$, we have

$$\begin{aligned} &\frac{p+qt}{i2\pi} + c + rt + (\frac{q}{i2\pi} + r)\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V_x^{\mathrm{I}}(t,\eta)} \\ &\quad = \frac{i2\pi V_{x\varphi'}^{\mathrm{I}}(t,\eta) - V_x^{I,g'}(t,\eta)}{i2\pi V_x^{\mathrm{I}}(t,\eta)} + \eta - \varphi'(t) - \eta_0. \end{aligned} \tag{29}$$

Taking partial derivative ∂_η to the both sides of (29), we have

$$(\frac{q}{i2\pi} + r)\partial_\eta\Big(\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V_x^{\mathrm{I}}(t,\eta)}\Big) = 1 + \partial_\eta\Big(\frac{i2\pi V_{x\varphi'}^{\mathrm{I}}(t,\eta) - V_x^{I,g'}(t,\eta)}{i2\pi V_x^{\mathrm{I}}(t,\eta)}\Big),$$

which leads to

$$\frac{q}{i2\pi} + r = Q_0(t,\eta),$$

for (t,η) with $\partial_\eta\big(V_x^{\mathrm{I},g_1}(t,\eta)/V_x^{\mathrm{I}}(t,\eta)\big) \neq 0$, where $Q_0(t,\eta)$ is defined by (27).

Returning back to (29) with $\frac{q}{i2\pi}+r$ replaced by $Q_0(t,\eta)$, we have

$$c+rt \;=\; \frac{i2\pi\, V^{\mathrm{I}}_{x\varphi'}(t,\eta)-V_x^{I,g'}(t,\eta)}{i2\pi\, V^{\mathrm{I}}_x(t,\eta)} \\ +\eta-\varphi'(t)-\eta_0-\frac{p+qt}{i2\pi}-Q_0(t,\eta)\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V^{\mathrm{I}}_x(t,\eta)}.$$

Since $c+rt$ is real, taking the real parts of the quantities in the above equation, we have

$$\begin{aligned} c+rt \;&= \mathrm{Re}\Big\{\frac{i2\pi\, V^{\mathrm{I}}_{x\varphi'}(t,\eta)-V_x^{I,g'}(t,\eta)}{i2\pi\, V^{\mathrm{I}}_x(t,\eta)}\Big\} \\ &\quad+\eta-\varphi'(t)-\eta_0-\mathrm{Re}\Big\{Q_0(t,\eta)\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V^{\mathrm{I}}_x(t,\eta)}\Big\} \\ &=\omega^{\mathrm{I}}_x(t,\eta)-\mathrm{Re}\Big\{Q_0(t,\eta)\frac{V_x^{\mathrm{I},g_1}(t,\eta)}{V^{\mathrm{I}}_x(t,\eta)}\Big\}, \end{aligned}$$

which is $\omega_x^{\mathrm{I},2\mathrm{nd}}(t,\eta)$. This completes the proof of Theorem 4. □

With the phase transformation $\omega_x^{\mathrm{I},2\mathrm{nd}}(t,\eta)$ in (26), we have the corresponding 2nd-order IFE-FSST of a signal $x(t)$ with φ, ξ_0 and window function g defined by

$$R_x^{\mathrm{I},2}(t,\xi) := \int_{\{\eta\,:\,V_x^{\mathrm{I}}(t,\eta)\neq 0\}} V^{\mathrm{I}}_x(t,\eta)\delta\big(\omega_x^{\mathrm{I},2\mathrm{nd}}(t,\eta)-\xi\big)d\eta. \tag{30}$$

One has reconstruction formulas with $R_x^{\mathrm{I},2}(t,\xi)$ similar to (22)–(25).

5. IMPLEMENTATION AND EXPERIMENTAL RESULTS

5.1. Calculating $\omega_x^{\mathrm{I}}(t,\eta)$ and $\omega_x^{\mathrm{I},2nd}(t,\eta)$

First we consider the IFE-FSST. We need to calculate $\omega^{\mathrm{I}}_x(t,\eta)$. We will use (15) so that FFT can be applied to (discrete signals) x and $x\varphi'$ to calculate $V^{\mathrm{I}}(t,\eta)$, $V^{\mathrm{I}}_{x\varphi'}(t,\eta)$ and $V_x^{\mathrm{I},g'}(t,\eta)$. $V^{\mathrm{I}}_{x\varphi'}(t,\eta)$ can be obtained by (15) with x replaced by $x\varphi'$. As long as $V_x^{\mathrm{I},g'}(t,\eta)$ is concerned, observe that the Fourier transform of g' is $i2\pi\xi\ \widehat{g}(\xi)$. Hence

$$V_x^{\mathrm{I},g'}(t,\eta) \;=\; e^{i2\pi\varphi(t)}\int_{\mathbb{R}}\widehat{x}(\xi)i2\pi(\eta-\varphi'(t)-\xi) \\ \widehat{g}(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}\,d\xi.$$

After obtaining $V^{\mathrm{I}}(t,\eta)$, $V^{\mathrm{I}}_{x\varphi'}(t,\eta)$ and $V_x^{\mathrm{I},g'}(t,\eta)$, we get $\omega^{\mathrm{I}}_x(t,\eta)$ and then the IFE-FSST.

For the 2nd-order IFE-FSST, we need to calculate

$$V_x^{\mathrm{I},g_1}(t,\eta), \partial_\eta\Big(V^{\mathrm{I}}_x(t,\eta)\Big), \partial_\eta\Big(V_x^{\mathrm{I},g_1}(t,\eta)\Big), \partial_\eta\Big(V^{\mathrm{I}}_{x\varphi'}(t,\eta)\Big), \\ \partial_\eta\Big(V_x^{\mathrm{I},g'}(t,\eta)\Big).$$

Note that the Fourier transform of $\tau g(\tau)$ is

$$\begin{aligned} \int_{\mathbb{R}}\tau g(\tau)e^{-i2\pi\xi\tau}d\tau \;&=\; \frac{1}{-i2\pi}\frac{d}{d\xi}\Big(\int_{\mathbb{R}}g(\tau)e^{-i2\pi\xi\tau}d\tau\Big) \\ &=\; \frac{1}{-i2\pi}\big(\widehat{g}\big)'(\xi). \end{aligned}$$

Thus, we conclude

$$V_x^{\mathrm{I},g_1}(t,\eta) = -e^{i2\pi\varphi(t)}\frac{1}{i2\pi}\int_{\mathbb{R}}\widehat{x}(\xi)\ \big(\widehat{g}\big)'(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}\,d\xi. \tag{31}$$

By the fact $\partial_\eta\Big(V^{\mathrm{I}}_x(t,\eta)\Big) \;=\; -i2\pi\, V_x^{\mathrm{I},g_1}(t,\eta)$, we can obtain $\partial_\eta\Big(V^{\mathrm{I}}_x(t,\eta)\Big)$ and $\partial_\eta\Big(V^{\mathrm{I}}_{x\varphi'}(t,\eta)\Big)$ as well *via* (31).

To calculate $\partial_\eta\big(V_x^{\mathrm{I},g_1}(t,\eta)\big)$, with $\partial_\eta\big(V_x^{\mathrm{I},g_1}(t,\eta)\big) = -i2\pi\, V_x^{\mathrm{I},g_2}(t,\eta)$, where $g_2(\tau)=\tau^2 g(\tau)$, we need to calculate the Fourier transform of $g_2(\tau)$, which is

$$\widehat{g_2}(\xi) \;=\; \frac{1}{(-i2\pi)^2}\frac{d^2}{d\xi^2}\Big(\int_{\mathbb{R}}g(\tau)e^{-i2\pi\xi\tau}d\tau\Big) = -\frac{1}{4\pi^2}\big(\widehat{g}\big)''(\xi).$$

Therefore,

$$\partial_\eta\Big(V_x^{\mathrm{I},g_1}(t,\eta)\Big) = -e^{i2\pi\varphi(t)}\frac{1}{i2\pi}\int_{\mathbb{R}}\widehat{x}(\xi)\ \big(\widehat{g}\big)''(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}\,d\xi. \tag{32}$$

For $\partial_\eta\Big(V_x^{\mathrm{I},g'}(t,\eta)\Big)$, we need to calculate the Fourier transform of $\tau g'(\tau)$, denoted by $\big(\tau g'(\tau)\big)^\wedge(\xi)$. Indeed,

$$\begin{aligned} \big(\tau g'(\tau)\big)^\wedge(\xi) \;&= \int_{\mathbb{R}}\tau g'(\tau)e^{-i2\pi\xi\tau}d\tau = \frac{1}{-i2\pi}\frac{d}{d\xi}\Big(\int_{\mathbb{R}}g'(\tau)e^{-i2\pi\xi\tau}d\tau\Big) \\ &= -\frac{1}{-i2\pi}\frac{d}{d\xi}\Big(\int_{\mathbb{R}}g(\tau)\partial_\tau\big(e^{-i2\pi\xi\tau}\big)d\tau\Big) \\ &= -\frac{d}{d\xi}\Big(\xi\int_{\mathbb{R}}g(\tau)e^{-i2\pi\xi\tau}d\tau\Big) \\ &= -\frac{d}{d\xi}\big(\xi\widehat{g}(\xi)\big) = -\widehat{g}(\xi)-\xi\big(\widehat{g}\big)'(\xi). \end{aligned}$$

Thus,

$$\begin{aligned} \partial_\eta\Big(V_x^{\mathrm{I},g'}(t,\eta)\Big) \;&= -i2\pi\, V_x^{\mathrm{I},\tau g'(\tau)}(t,\eta) \\ &= -i2\pi\, e^{i2\pi\varphi(t)}\int_{\mathbb{R}}\widehat{x}(\xi)\ \big(\tau g'(\tau)\big)^\wedge(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}\,d\xi \\ &= i2\pi\, V^{\mathrm{I}}(t,\eta)+i2\pi\, e^{i2\pi\varphi(t)} \\ &\quad\int_{\mathbb{R}}\widehat{x}(\xi)\ (\eta-\varphi'(t)-\xi)\big(\widehat{g}\big)'(\eta-\varphi'(t)-\xi)e^{i2\pi t\xi}\,d\xi. \end{aligned} \tag{33}$$

With the formulas (31), (32), and (33), we can obtain $Q_0(t,\eta)$ and then, $\omega_x^{\mathrm{I},2nd}(t,\eta)$.

5.2. IFE-FSST Algorithms for IF Estimation and Component Recovery and Experiments

To apply IFE-STFT or IFE-FSST, first of all we need to choose $\varphi(t)$ and $\varphi'(t)$. For the purpose of estimating the IF $\phi_k'(t)$ of the kth component $x_k(t)$ and/or recover $x_k(t)$ of a multicomponent signal $x(t)$, we should choose $\varphi(t)$ and $\varphi'(t)$ close to $\phi_k(t)$ (up to a constant) and $\phi_k'(t)$ respectively. One way is to use the ridges of the STFT. More precisely, suppose $\{t_n\}_{0\le n<N}, \{\eta_j\}_{0\le j<J}, \{\xi_m\}_{0\le m<M}$ are the sampling points of t, η, ξ respectively for STFT $V_x(t,\eta)$, FSST $R_x(t,\xi)$, and IFE-FSST $R_x^{\mathrm{I}}(t,\xi)$. Let $\widehat{\eta}_{j_n,k}, 0 \le n < N$ be the STFT ridge corresponding to $x_k(t)$ given by

$$\widehat{\eta}_{j_n,k} := \mathrm{argmax}_{\eta_j \in \mathcal{G}_{t_n,k}}\{|V_x(t_n,\eta_j)|\}, \tag{34}$$

for each n, $0 \le n < N$, where for each n, $\mathcal{G}_{t_n,k}$ is an interval containing $\phi_k'(t_n)$ (with convention: $\phi_0(t) \equiv 0$) at the time instant t_n, and $\mathcal{G}_{t_n,k}, 0 \le k \le K$ form a disjoint union of $\{\eta : |V_x(t_n,\eta)| > \gamma\}$, namely for each t_n,

$$\{\eta : |V_x(t_n,\eta)| > \gamma\} = \cup_{k=0}^{K}\mathcal{G}_{t_n,k}.$$

See more details on $\mathcal{G}_{t,k}$ in [37].

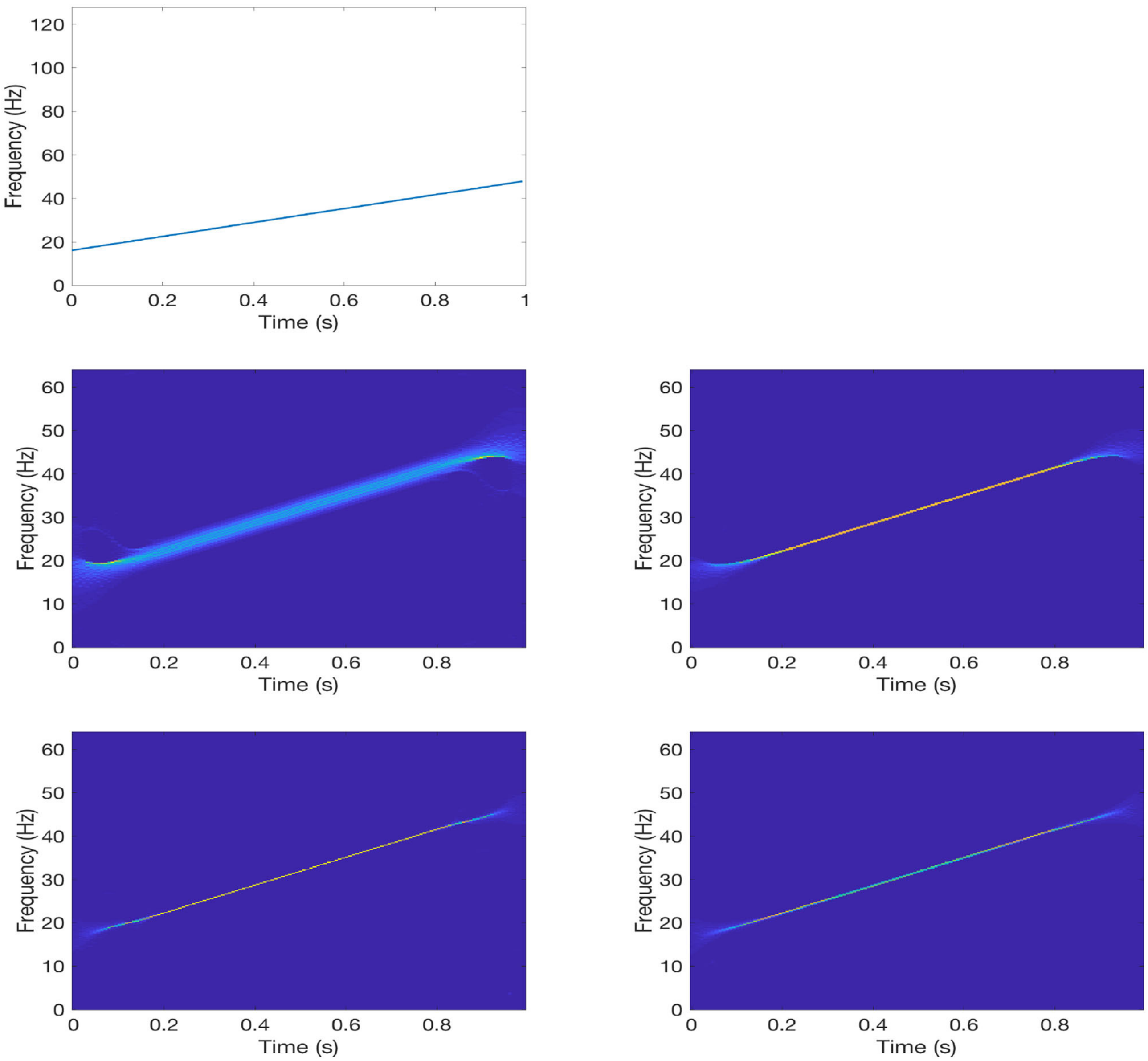


FIGURE 1 | Experiment with $x(t)$ in (43). **1st row:** IF $\phi'(t)$; **2nd row:** FSST $|R_x(t,\eta)|$ **(left)** and IFE-FSST $|R_x^{\mathrm{I}}(t,\eta)|$ **(right)**; **3rd row:** 2nd-order FSST **(left)** and 2nd-order IFE-FSST $|R_x^{\mathrm{I},2}(t,\eta)|$ **(right)**.

$\{\widehat{\eta}_{j_n,k}\}_{n=0}^{N-1}$ is called a ridge of the STFT plane or a ridge of the spectrogram $|V_x(t,\eta)|$. It provides an approximation to $\phi_k'(t_n), 0\le n<N$ [see [36, 37, 45]]. Thus, we can use

$$\varphi'(t_n)=\widehat{\eta}_{j_n,k},\ \varphi(t_n)=\sum_{\ell=0}^{n-1}\widehat{\eta}_{j_\ell,k}\triangle t_\ell,\ 0\le n<N \tag{35}$$

as discrete $\varphi'(t)$ and $\varphi(t)$ to define IFE-STFT and IFE-FSST, where $\triangle t_\ell=t_\ell-t_{\ell-1}$.

To recover a component by either FSST or IFE-FSST, we need an estimate $\mathrm{IF}_k(t)$ for $\phi_k'(t)$ so that (10) or (24)/(25) can be applied. One way is to use the ridges of FSST and IFE-FSST to approximate $\phi_k'(t_n)$. More precisely, let $\widehat{\xi}_{m_n,k}, 0\le n<N$ be the FSST ridge defined similarly to the STFT ridge in (34):

$$\widehat{\xi}_{m_n,k}:=\mathrm{argmax}_{\xi_m\in\mathcal{G}_{t_n,k}}\{|R_x(t_n,\xi_m)|\},\quad 0\le n<N. \tag{36}$$

Then Equation (10) becomes

$$x_k(t_n)\approx x_k^{\rm rec}(t_n):=\frac{1}{g(0)}\sum_{\{m:\ |m-m_n|<M_0\}}R_x(t_n,\xi_m)\triangle\xi_m,\quad 0\le n<N, \tag{37}$$

for some $M_0\in\mathbb{N}$, where $\triangle\xi_m=\xi_m-\xi_{m-1}$.

Similarly, Equation (24) implies that $x_k(t)$ can be recovery from (discrete) IFE-FSST:

$$x_k(t_n)\approx x_k^{\rm I,rec}(t_n):=\frac{e^{-i2\pi\eta_0t_n}}{g(0)}\sum_{\{m:\ |m-m_n^{\rm I}|<M_0\}}R_x^{\rm I}(t_n,\xi_m)\triangle\xi_m,\quad 0\le n<N, \tag{38}$$

where $m_n^{\rm I}, 0\le n<N$ are the indices for IFE-FSST ridge defined as (36) with $R_x(t_n,\xi_m)$ replaced by $R_x^{\rm I}(t_n,\xi_m)$:

$$\widehat{\xi}_{m_n^{\rm I},k}:=\mathrm{argmax}_{\xi_m\in\mathcal{G}_{t_n,k}}\{|R_x^{\rm I}(t_n,\xi_m)|\},\quad 0\le n<N. \tag{39}$$

For real-valued $x_k(t)$ and $g(t)$, the recovery formulas (37) and (38) are respectively

$$x_k(t_n)\approx x_k^{\rm rec}(t_n):=\frac{2}{g(0)}\mathrm{Re}\Big(\sum_{\{m:\ |m-m_n|<M_0\}}R_x(t_n,\xi_m)\triangle\xi_m\Big),\quad 0\le n<N, \tag{40}$$

and

$$x_k(t_n)\approx x_k^{\rm I,rec}(t_n) \tag{41}$$

$$:=\frac{2}{g(0)}\mathrm{Re}\Big(e^{-i2\pi\eta_0t_n}\sum_{\{m:\ |m-m_n^{\rm I}|<M_0\}}R_x^{\rm I}(t_n,\xi_m)\triangle\xi_m\Big),\quad 0\le n<N. \tag{42}$$

To summarize, we have the following algorithm to estimate IF $\phi_k'(t)$ and recover $x_k(t)$ by IFE-FSST.

Algorithm 1. (IFE-FSST algorithm for IF estimation and component recovery) *Let $x(t)$ be a signal of the form (1). To estimate $\phi_k'(t)$ and recover $x_k(t)$ by IFE-FSST, do the following.*

Step 1. *Obtain the STFT ridge $\widehat{\eta}_{j_n,k}, 0\le n<N$ by (34) and $\varphi'(t_n),\varphi(t_n), 0\le n<N$ by (35).*
Step 2. *Calculate IFE-FSST with φ',φ obtained in Step 1. The ridge $\widehat{\xi}_{m_n^{\rm I},k}, 0\le n<N$ defined by (39) is an estimate of $\phi_k'(t)$ and $x_k^{\rm I,rec}(t)$ in (38) is an approximation to $x_k(t)$.*

We can use **Algorithm 1** to recover each component $x_k(t)$ one by one. We can also apply **Algorithm 1** to the remainder $x(t)-x_k^{\rm I,rec}(t)$ to recover another component after $x_k(t)$ is recovered; and we can repeat this procedure. The procedure of this iterative method is described as follows.

Algorithm 2. (Iterative IFE-FSST algorithm for IF estimation and component recovery) *Let $x(t)$ be a signal of the form (1).*

Step 1. *Apply **Algorithm 1** to obtain $x_1^{\rm I,rec}(t)$.*
Step 2. *Let $y_1=x-x_1^{\rm I,rec}$. Apply **Algorithm 1** to y_1 to obtain $x_2^{\rm I,rec}(t)$.*
Step 3. *Let $y_2=x-x_1^{\rm I,rec}-x_2^{\rm I,rec}$. Apply **Algorithm 1** to y_2 to obtain $x_3^{\rm I,rec}(t)$. Repeat this process to obtain $x_4^{\rm I,rec}(t),\cdots$, and finally $x_K^{\rm I,rec}(t)$.*

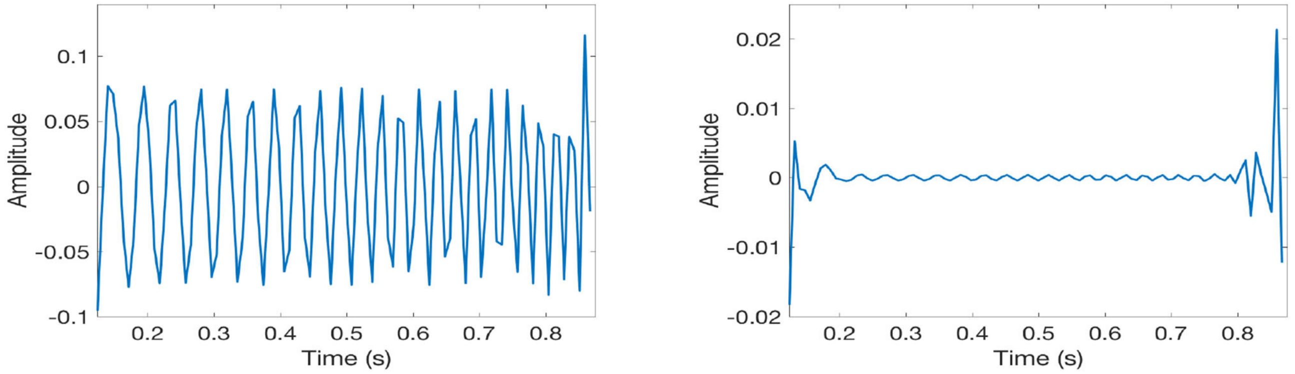


FIGURE 2 | Recovery errors for *x*(*t*) given in (43) on [0.125, 0.875] by FSST **(left)** and IFE-FSST **(right)**.

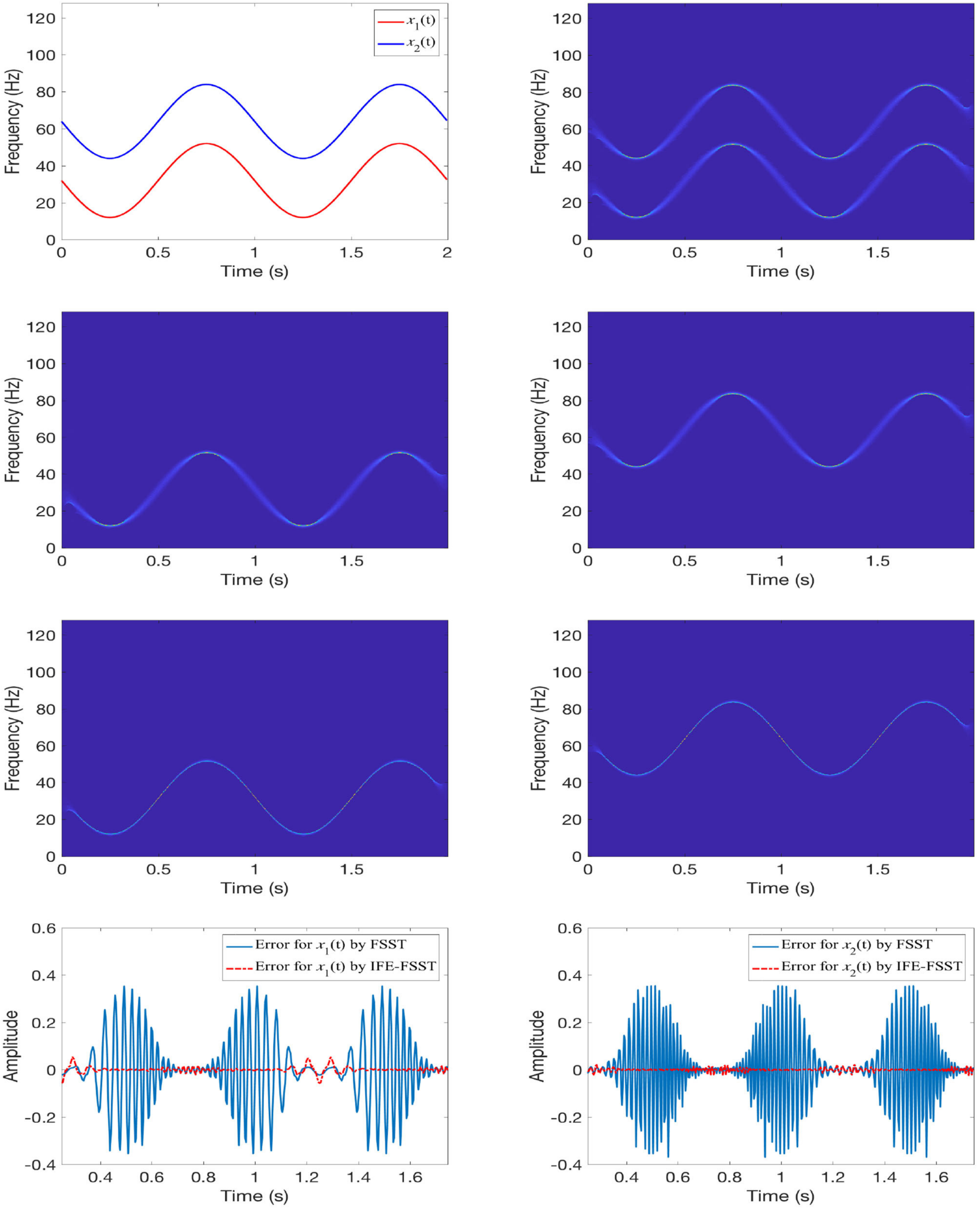


FIGURE 3 | Experiment with $x(t)$ in (44). **1st row:** IFs $\phi_1'(t)$, $\phi_2'(t)$ **(left)** and FSST $|R_x(t,\eta)|$ **(right)**; **2nd row:** FSST for $x_1(t)$ **(left)** and FSST for $x_2(t)$ **(right)**; **3rd row:** IFE-FSST for $x_1(t)$ **(left)** and IFE-FSST for $x_2(t)$ **(right)**. **4th row:** recovery errors on [0.125, 1.875) by FSST and IFE-FSST for $x_1(t)$ **(left)** and $x_2(t)$ **(right)**.

Step 4. *Apply **Algorithm 1** to $x - \sum_{k=2}^{K} x_k^{\mathrm{I,rec}}$ to recover $x_1(t)$. Let $\widetilde{x}_1^{\mathrm{I,rec}}(t)$ be the recovered $x_1(t)$. Then Apply **Algorithm 1** to $x - \widetilde{x}_1^{\mathrm{I,rec}}(t) - \sum_{k=3}^{K} x_k^{\mathrm{I,rec}}$ to recover $x_2(t)$. Let $\widetilde{x}_2^{\mathrm{I,rec}}(t)$ be the recovered $x_2(t)$. Obtain $\widetilde{x}_3^{\mathrm{I,rec}}(t)$ by applying **Algorithm 1** to $x - \widetilde{x}_1^{\mathrm{I,rec}}(t) - \widetilde{x}_2^{\mathrm{I,rec}}(t) - \sum_{k=4}^{K} x_k^{\mathrm{I,rec}}$. Repeat this process to obtain $\widetilde{x}_4^{\mathrm{I,rec}}(t), \cdots$, and finally $\widetilde{x}_K^{\mathrm{I,rec}}(t)$.*

We can repeat the procedure in Step 4 of **Algorithm 2**. That is why we call **Algorithm 2** an iterative algorithm.

Next we consider two examples. We let

$$g(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{t^2}{2\sigma^2}},$$

be the window function, where $\sigma > 0$. First we consider a mono-component signal

$$x(t) = \cos\left(2\pi(\phi(t)\right) = \cos\left(2\pi(16t + 16t^2)\right),\ t \in [0, 1), \quad (43)$$

where $x(t)$ is uniformly sampled with sample points $t_n = n\triangle t, 0 \le n < N = 128, \triangle t = \frac{1}{128}$. The IF of $x(t)$ is $\phi'(t) = 16 + 32t$ and it is shown in the 1st row of **Figure 1**. The FSST and IFE-FSST of $x(t)$ are provided in the 2nd row; and the 2nd-order FSST and IFE-FSST are shown in the 3rd row. In this example we let $\sigma = \frac{1}{16}$. As mentioned above, discrete $\varphi'(t)$ and $\varphi(t)$ defined by (35) are used to define IFE-STFT and the 2nd-order IFE-STFT. Obviously IFE-FSST provides a much sharper time-frequency representation of $x(t)$ than FSST. Both the 2nd-order FSST and the 2nd-order IFE-FSST as well give sharp time-frequency representations of $x(t)$.

For a mono-component signal $x(t)$ as given by (43), since $x(t)$ can be recovered from FSST or IFE-FSST as shown in (8) and (22) respectively, theoretically, either (40) or (41) gives high accurate approximation to $x(t)$ as long as M_0 is large enough. We choose a small M_0 so that the recovery errors with it show how sharp the time-frequency representations with FSST and IFE-FSST are. Here and below we set $M_0 = 8$.

In **Figure 2**, we provide the recovery errors $x^{\mathrm{rec}}(t_n) - x(t_n)$, $x^{\mathrm{I,rec}}(t_n) - x(t_n)$ for $x(t)$ by FSST and IFE-FSST, where $x^{\mathrm{rec}}(t_n)$ and $x^{\mathrm{I,rec}}(t_n)$ are given by (40) and (41) respectively with $M_0 = 8$. Here, we show the error on $[0.125, 0.875)$ only to ignore the boundary effect. Obviously, IFE-FSST provides a much sharper time-frequency representation than FSST.

Next we consider a two-component signal given by

$$x(t) = x_1(t) + x_2(t),\ x_1(t) = \cos\left(2\pi\left(32t + \frac{10}{\pi}\cos(2\pi t)\right)\right),$$
$$x_2(t) = \cos\left(2\pi\left(64t + \frac{10}{\pi}\cos(2\pi t)\right)\right), \quad (44)$$

where $t \in [0, 2)$, and $x(t)$ is uniformly sampled with sample points $t_n = n\triangle t, 0 \le n < N = 512, \triangle t = \frac{1}{256}$. Thus, IFs of $x_1(t), x_2(t)$ are $\phi_1'(t) = 32 - 20\sin(2\pi t), \phi_2'(t) = 64 - 20\sin(2\pi t)$, which are shown on the top-left panel of **Figure 3**. In this example we let $\sigma = \frac{1}{32}$ for the window function.

To this two-component signal, we apply **Algorithm 2** to obtain $\widetilde{x}_1^{\mathrm{I,rec}}(t)$ and $\widetilde{x}_2^{\mathrm{I,rec}}(t)$. In the 3rd row of **Figure 3** we show the IFE-FSSTs of $\widetilde{x}_1^{\mathrm{I,rec}}(t)$ and $\widetilde{x}_2^{\mathrm{I,rec}}(t)$. The FSST of $x(t)$ is provided in the top-right panel of **Figure 3**. Of course, we can also apply iterative method to FSST to recover components one by one. Namely, we apply FSST to obtain $x_1^{\mathrm{rec}}(t)$, then apply FSST to $x(t) - x_1^{\mathrm{rec}}(t)$ to obtain $x_2^{\mathrm{rec}}(t)$. After that we apply FSST to $x(t) - x_2^{\mathrm{rec}}(t)$ to obtain $\widetilde{x}_1^{\mathrm{rec}}(t)$, and finally to obtain $\widetilde{x}_2^{\mathrm{rec}}(t)$ by applying FSST to $x(t) - \widetilde{x}_1^{\mathrm{rec}}(t)$. The 2nd row of **Figure 3** shows the FSSTs of $\widetilde{x}_1^{\mathrm{rec}}(t)$ and $\widetilde{x}_2^{\mathrm{rec}}(t)$. Comparing the FSST of x in the top-right panel with the individual FSSTs in the 2nd row, we see there is not much improvement of the time-frequency representation of FSST of x after we apply the iterative component recovery procedure.

In the 4th row of **Figure 3**, we provide the recovery errors for $x_1(t), x_2(t)$ by FSST and IFE-FSST. Here, we show the error on $[0.125, 1.875)$. From **Figure 3**, we see IFE-FSST provides a much sharper time-frequency representation for $x(t)$. We also consider FSST and IFE-FSST of two-component $x(t)$ in the noisy environment and our experiments show that IFE-FSST provides a sharp time-frequency representation in the noisy environment. In addition, we consider the 2nd-order IFE-FSST for component recovery. It does not provide much improvement than IFE-FSST. This may be due to that the results from IFE-FSST are hard to improve. Due to that only 15 pictures are allowed to be included in a article in this journal, we do not present these results here.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

ETHICS STATEMENT

This article was approved by AFRL for public release on 03 Dec. 2021, Case Number: AFRL-2021-4285, Distribution unlimited.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

This work was supported in part by Simons Foundation (Grant No. 353185) and the 2020 Air Force Summer Faculty Fellowship Program (SFFP).

REFERENCES

1. Daubechies I, Lu J, Wu H-T. Synchrosqueezed wavelet transforms: an empirical mode decomposition-like tool. *Appl Comput Harmon Anal*. (2011) 30:243–61. doi: 10.1016/j.acha.2010.08.002
2. Huang NE, Shen Z, Long SR, Wu ML, Shih HH, Zheng Q, et al. The empirical mode decomposition and Hilbert spectrum for nonlinear and nonstationary time series analysis. *Proc R Soc Lond A*. (1998) 454:903–95. doi: 10.1098/rspa.1998.0193
3. Wu Z, Huang NE. Ensemble empirical mode decomposition: a noise-assisted data analysis method. *Adv Adapt Data Anal*. (2009) 1:1–41. doi: 10.1142/S1793536909000047
4. Flandrin P, Rilling G, Goncalves P. Empirical mode decomposition as a filter bank. *IEEE Signal Proc Lett*. (2004) 11:112–4. doi: 10.1109/LSP.2003.821662
5. Li L, Ji H. Signal feature extraction based on improved EMD method. *Measurement*. (2009) 42:796–803. doi: 10.1016/j.measurement.2009.01.001
6. Rilling G, Flandrin P. One or two frequencies? The empirical mode decomposition answers. *IEEE Trans Signal Proc*. (2008) 56:85–95. doi: 10.1109/TSP.2007.906771
7. Lin L, Wang Y, Zhou HM. Iterative filtering as an alternative algorithm for empirical mode decomposition. *Adv Adapt Data Anal*. (2009) 1:543–60. doi: 10.1142/S179353690900028X
8. Xu Y, Liu B, Liu J, Riemenschneider S. Two-dimensional empirical mode decomposition by finite elements. *Proc R Soc Lond A*. (2006) 462:3081–96. doi: 10.1098/rspa.2006.1700
9. van der Walt MD. Empirical mode decomposition with shape-preserving spline interpolation. *Results Appl Math*. (2020) 5:100086. doi: 10.1016/j.rinam.2019.100086
10. Wang Y, Wei G-W, Yang SY. Iterative filtering decomposition based on local spectral evolution kernel. *J Sci Comput*. (2012) 50:629–64. doi: 10.1007/s10915-011-9496-0
11. Zheng JD, Pan HY, Liu T, Liu QY. Extreme-point weighted mode decomposition. *Signal Proc*. (2018) 42:366–74. doi: 10.1016/j.sigpro.2017.08.00
12. Cicone A, Liu JF, Zhou HM. Adaptive local iterative filtering for signal decomposition and instantaneous frequency analysis. *Appl Comput Harmon Anal*. (2016) 41:384–411. doi: 10.1016/j.acha.2016.03.001
13. Thakur G, Wu H-T. Synchrosqueezing based recovery of instantaneous frequency from nonuniform samples. *SIAM J Math Anal*. (2011) 43:2078–95. doi: 10.1137/100798818
14. Wu H-T. *Adaptive analysis of complex data sets*. Ph.D. dissertation. Princeton University Press, Princeton, NJ, United States (2012).
15. Oberlin T, Meignen S, Perrier V. The Fourier-based synchrosqueezing transform. In: *Proc. 39th Int. Conf. Acoust., Speech, Signal Proc.* (*ICASSP*). Beijing (2014). p. 315–9. doi: 10.1109/ICASSP.2014.6853609
16. Oberlin T, Meignen S, Perrier V. Second-order synchrosqueezing transform or invertible reassignment? Towards ideal time-frequency representations. *IEEE Trans Signal Proc*. (2015) 63:1335–44. doi: 10.1109/TSP.2015.2391077
17. Oberlin T, Meignen S. The 2nd-order wavelet synchrosqueezing transform. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. New Orleans, LA (2017). doi: 10.1109/ICASSP.2017.7952906
18. Lu J, Alzahrani JH, Jiang QT. A second-order synchrosqueezing transform with a simple phase transformation. *Num Math Theory Methods Appl*. (2021) 14: 624–49. doi: 10.4208/nmtma.OA-2020-0077
19. Pham D-H, Meignen S. High-order synchrosqueezing transform for multicomponent signals analysis - With an application to gravitational-wave signal. *IEEE Trans Signal Proc*. (2017) 65:3168–78. doi: 10.1109/TSP.2017.2686355
20. Li L, Wang ZH, Cai HY, Jiang QT, Ji HB. Time-varying parameter-based synchrosqueezing wavelet transform with the approximation of cubic phase functions. In: *2018 14th IEEE Int'l Conference on Signal Proc. ICSP*. New Orleans, LA (2018). p. 844–8. doi: 10.1109/ICSP.2018.8652362
21. Chui CK, van der Walt MD. Signal analysis via instantaneous frequency estimation of signal components. *Int J Geomath*. (2015) 6:1–42. doi: 10.1007/s13137-015-0070-z
22. Chui CK, Lin Y-T, Wu H-T. Real-time dynamics acquisition from irregular samples - with application to anesthesia evaluation. *Anal Appl*. (2016) 14:537–90. doi: 10.1142/S0219530515500165
23. Daubechies I, Wang Y, Wu H-T. ConceFT: concentration of frequency and time via a multitapered synchrosqueezed transform. *Philos Trans R Soc A*. (2016) 374:20150193. doi: 10.1098/rsta.2015.0193
24. Yang HZ. Synchrosqueezed wave packet transforms and diffeomorphism based spectral analysis for 1D general mode decompositions. *Appl Comput Harmon Anal*. (2015) 39:33–66. doi: 10.1016/j.acha.2014.08.004
25. Yang HZ, Ying LX. Synchrosqueezed curvelet transform for two-dimensional mode decomposition. *SIAM J. Math Anal*. (2014) 3:2052–83. doi: 10.1137/130939912
26. Sheu Y-L, Hsu L-Y, Chou P-T, Wu H-T. Entropy-based time-varying window width selection for nonlinear-type time-frequency analysis. *Int J Data Sci Anal*. (2017) 3:231–45. doi: 10.1007/s41060-017-0053-2
27. Berrian AJ, Saito N. Adaptive synchrosqueezing based on a quilted short-time Fourier transform. *arXiv [Preprint] arXiv*:1707.03138v5. (2017). doi: 10.1117/12.2271186
28. Li L, Cai HY, Han HX, Jiang QT, Ji HB. Adaptive short-time Fourier transform and synchrosqueezing transform for non-stationary signal separation. *Signal Proc*. (2020) 166:107231. doi: 10.1016/j.sigpro.2019.07.024
29. Li L, Cai HY, Jiang QT. Adaptive synchrosqueezing transform with a time-varying parameter for non-stationary signal separation. *Appl Comput Harmon Anal*. (2020) 49:1075–106. doi: 10.1016/j.acha.2019.06.002
30. Cai HY, Jiang QT, Li L, Suter BW. Analysis of adaptive short-time Fourier transform-based synchrosqueezing transform. *Anal Appl*. (2021) 19:71–105. doi: 10.1142/S0219530520400047
31. Lu J, Jiang QT, Li L. Analysis of adaptive synchrosqueezing transform with a time-varying parameter. *Adv Comput Math*. (2020) 46:72. doi: 10.1007/s10444-020-09814-x
32. Li C, Liang M. Time frequency signal analysis for gearbox fault diagnosis using a generalized synchrosqueezing transform. *Mech Syst Signal Proc*. (2012) 26:205–17. doi: 10.1016/j.ymssp.2011.07.001
33. Wang SB, Chen XF, Selesnick IW, Guo YJ, Tong CW, Zhang XW. Matching synchrosqueezing transform: a useful tool for characterizing signals with fast varying instantaneous frequency and application to machine fault diagnosis. *Mech Syst Signal Proc*. (2018) 100:242–88. doi: 10.1016/j.ymssp.2017.07.009
34. Wu H-T. Current state of nonlinear-type time-frequency analysis and applications to high-frequency biomedical signals. *Curr Opin Syst Biol*. (2020) 23:8–21. doi: 10.1016/j.coisb.2020.07.013
35. Chui CK, Mhaskar HN. Signal decomposition and analysis via extraction of frequencies. *Appl Comput Harmon Anal*. (2016) 40:97–136. doi: 10.1016/j.acha.2015.01.003
36. Li L, Chui CK, Jiang QT. Direct signal separation via extraction of local frequencies with adaptive time-varying parameter. *arXiv [Preprint] arXiv*:2010.01866. (2020).
37. Chui CK, Jiang QT, Li L, Lu J. Analysis of an adaptive short-time Fourier transform-based multicomponent signal separation method derived from linear chirp local approximation. *J Comput Appl Math*. (2021) 396:113607. doi: 10.1016/j.cam.2021.113607
38. Chui CK, Han NN. Wavelet thresholding for recovery of active sub-signals of a composite signal from its discrete samples. *Appl Comput Harmon Anal*. (2021) 52:1–24. doi: 10.1016/j.acha.2020.11.003
39. Chui CK, Jiang QT, Li L, Lu J. Signal separation based on adaptive continuous wavelet-like transform and analysis. *Appl Comput Harmon Anal*. (2021) 53:151–79. doi: 10.1016/j.acha.2020.12.003
40. Chui CK, Mhaskar HN, van der Walt MD. Data-driven atomic decomposition via frequency extraction of intrinsic mode functions. *Int J Geomath*. (2016) 7:117–46. doi: 10.1007/s13137-015-0079-3
41. Li C, Liang M. A generalized synchrosqueezing transform for enhancing signal time-frequency representation. *Signal Proc*. (2012) 92:2264–74. doi: 10.1016/j.sigpro.2012.02.019
42. Meignen S, Pham D-H, McLaughlin S. On demodulation, ridge detection, and synchrosqueezing for multicomponent signals. *IEEE Trans Signal Proc*. (2017) 65:2093–103. doi: 10.1109/TSP.2017.2656838

43. Wang SB, Chen XF, Cai GG, Chen BQ, Li X, He ZJ. Matching demodulation transform and synchrosqueezing in time-frequency analysis. *IEEE Trans Signal Proc*. (2014) 62:69–84. doi: 10.1109/TSP.2013.2276393
44. Jiang QT, Suter BW. Instantaneous frequency estimation based on synchrosqueezing wavelet transform. *Signal Proc*. (2017) 138:167–81. doi: 10.1016/j.sigpro.2017.03.007
45. Stankovic L, Dakovic M, Ivanovic V. Performance of spectrogram as IF estimator. *Electron Lett*. (2001) 37:797–9. doi: 10.1049/el:20010517

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

frontiers
in Applied Mathematics and Statistics

METHODS
published: 25 March 2022
doi: 10.3389/fams.2022.801830

Coupling Power Laws Offers a Powerful Modeling Approach to Certain Prediction/Estimation Problems With Quantified Uncertainty

Zhanshan (Sam) Ma [1,2]*

[1] *Computational Biology and Medical Ecology Lab, Kunming Institute of Zoology, Chinese Academy of Sciences, Kunming, China,* [2] *Center for Excellence in Animal Genetics and Evolution, Chinese Academy of Sciences, Kunming, China*

Power laws (PLs) have been found to describe a wide variety of natural (physical, biological, astronomic, meteorological, and geological) and man-made (social, financial, and computational) phenomena over a wide range of magnitudes, although their underlying mechanisms are not always clear. In statistics, PL distribution is often found to fit data exceptionally well when the normal (Gaussian) distribution fails. Nevertheless, predicting PL phenomena is notoriously difficult because of some of its idiosyncratic properties, such as lack of well-defined average value and potentially unbounded variance. Taylor's power law (TPL) is a PL first discovered to characterize the spatial and/or temporal distribution of biological populations. It has also been extended to describe the spatiotemporal heterogeneities (distributions) of human microbiomes and other natural and artificial systems, such as fitness distribution in computational (artificial) intelligence. The PL with exponential cutoff (PLEC) is a variant of power-law function that tapers off the exponential growth of power-law function ultimately and can be particularly useful for certain predictive problems, such as biodiversity estimation and turning-point prediction for Coronavirus Diease-2019 (COVID-19) infection/fatality. Here, we propose coupling (integration) of TPL and PLEC to offer a methodology for quantifying the uncertainty in certain estimation (prediction) problems that can be modeled with PLs. The coupling takes advantage of variance prediction using TPL and asymptote estimation using PLEC and delivers CI for the asymptote. We demonstrate the integrated approach to the estimation of potential (dark) biodiversity of the American gut microbiome (AGM) and the turning point of COVID-19 fatality. We expect this integrative approach should have wide applications given duel (contesting) relationship between PL and normal statistical distributions. Compared with the worldwide COVID-19 fatality number on January 24th, 2022 (when this paper is online), the error rate of the prediction with our coupled power laws, made in the May 2021 (based on the fatality data then alone), is approximately 7% only. It also predicted that the turning (inflection) point of the worldwide COVID-19 fatality would not occur until the July of 2022, which contrasts with a recent prediction made by Murray on January 19th of 2022, who suggested that the "end of the pandemic is near" by March 2022.

Keywords: Taylor's power law (TPL), power law with exponential cutoff (PLEC), potential (dark) biodiversity, long-tail skewed distribution, turning point of COVID-19, COVID-19 fatality prediction

OPEN ACCESS

Edited by:
Jun Ma,
Lanzhou University of Technology, China

Reviewed by:
Ilaria Spassiani,
Istituto Nazionale di Geofisica e Vulcanologia (INGV), Italy
Lijun Pei,
Zhengzhou University, China

****Correspondence:***
Zhanshan (Sam) Ma
ma@vandals.uidaho.edu

Specialty section:
This article was submitted to Dynamical Systems, a section of the journal Frontiers in Applied Mathematics and Statistics

Received: *26 October 2021*
Accepted: *24 January 2022*
Published: *25 March 2022*

Citation:
Ma ZS (2022) Coupling Power Laws Offers a Powerful Modeling Approach to Certain Prediction/Estimation Problems With Quantified Uncertainty. Front. Appl. Math. Stat. 8:801830. doi: 10.3389/fams.2022.801830

INTRODUCTION

A power law (PL) describes a non-linear functional relationship between two variables—one varies as a power of another (e.g., $f(x) = ax^b$) and has certain properties, such as scale invariance, lack of well-defined average value, and universality [1–4]. The scale invariance is exhibited by a simple log-transformation of PL into a straight-line (linear) on log-log scale {e.g., $\ln[f(x)] = \ln(a) + b\ln(x)$}, and it also specifies that all PLs with a particular scaling exponent are equivalent up to constant factors, e.g., $f(cx) = a(cx)^b = c^b f(x) \propto f(x)$. The lack of well-defined average value refers to a reality that arithmetic mean or average is a poor indicator for the majority of the power-law variables (e.g., the average income of a population that includes a billionaire). A PL usually has a well-defined mean only for a certain range of its scaling exponents, and the variance of PL seems disproportionally large and is frequently not well defined, which explains the association between PL phenomena and black swan behavior. This also makes many classic statistical methods that are based on the normal distribution and/or on the homogeneity of variance inapplicable to data of PL phenomena. The third property of PL is the universality that is to do with the scale invariance or the equivalence of PLs with a particular scaling exponent. In dynamic systems, diverse systems with the same power-law scaling exponents (also known as critical exponents) can exhibit identical scaling behavior and share the same fundamental dynamics as they approach criticality, such as phase transitions. Systems with the same critical exponents are classified as belonging to the same universality class [1–6].

Taylor's power law (TPL), first discovered by entomologist and ecologist L. R. Taylor [7] in investigating the spatial distribution of insect populations more than a half-century ago [5, 8–12], has been expanded far beyond its original domains of agricultural entomology and population ecology [1, 2, 5, 6, 13–19]. The TPL is one form of PLs that describe the distributions of a wide variety of natural and man-made phenomena over a wide range of scales [20–22]. PL patterns have been discovered/rediscovered in astronomy, biology and ecology, computer science, criminology, economics, finance, geology, mathematics, meteorology, physics, statistics, and especially in inter-disciplinary fields [3, 4].

Taylor's power law, as one of the most well-known PLs in ecology and biology, shares the three general properties of PLs mentioned above. It differs from other PLs in choosing its two variables: the mean (M) and variance (V) of population abundances (counts) [5, 7, 11], i.e.,$V = aM^b$. It has also been rediscovered in many other fields beyond its original domain of population ecologies, such as epidemiology, genomics and metagenomics, and computer science [5, 6, 14, 15, 20–23]. It was extended to community ecology, especially the community and landscape ecology of human microbiomes [6, 23, 24]. In the present study, we take the advantage of TPL in modeling the relationship between mean and variance for quantifying the uncertainty of natural phenomenon. This should be feasible because variance is arguably the most commonly used statistic moment for characterizing the uncertainty (variation) of random variables. The approach is particularly advantageous if the distribution of random variable follows PL distribution, but it should still be applicable otherwise since TPL holds across a wide range of mean-variance relationships as signaled by a wide range of its scaling parameter (b).

Species-area relationship (SAR) is another classic PL in ecology, which relates the number of species (species richness: S) and the area (A) of species habitat, in the form of $S = cA^z$. Ma [25, 26] further extended the SAR to a general diversity-area relationship (DAR) by replacing species number (richness) with the general diversity measured in Hill numbers. Ma [25–27] further introduced PL with exponential cutoff (PLEC) model to describe DAR and proposed the concept of maximal accrual diversity (MAD). Based on the PLEC model for DAR, Ma derived the estimation of MAD. MAD can be considered as a proxy of potential (dark) diversity, which includes both local diversity and the portion of diversity that are absent locally but present regionally (or in regional species pools). In other words, potential diversity measures both visible and invisible (dark) diversities and is of obvious significance for biodiversity conservation. Similar to SAR/DAR, there is so-called species-time relationship (STR) or diversity-time relationship (DTR) [26]. The PLEC version of DTR was successfully applied to predict the inflection points (tipping points) of COVID-19 infections [28].

Power law with exponential cutoff, as a variant of PL, has more general applications beyond the abovementioned SAR/DAR/STR/DTR/COVID-19 predictions [25–28]. PL behaves (grows or declines) exponentially, especially at late stages, and the PLEC possesses an exponential cutoff parameter that ultimately tapers off the unlimited growth or decline ultimately. Therefore, the PLEC model is of important practical significance when prediction or estimation is needed. However, existing PLEC modeling can only provide point estimation and not the interval of the estimation (i.e., uncertainty quantification of the estimation).

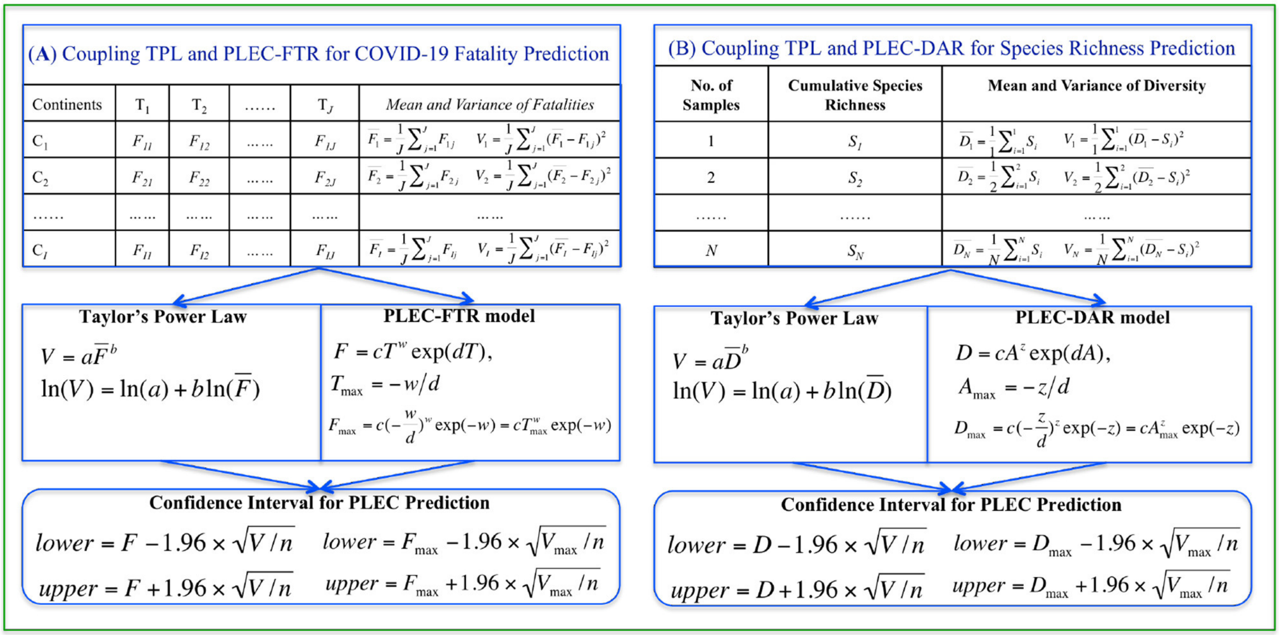
The present article is aimed to integrate the TPL with the PLEC model with the objective to improve the predictive power of the PLEC model by quantifying the uncertainty of estimation (prediction) with TPL. Specifically, by harnessing the capacity of TPL in estimating the variance (SD), we develop an approach to offering CIs for the estimation of PLEC quantities (see **Figure 1**). We demonstrate our method with the estimations of potential American gut microbiome (AGM) diversity and COVID-19 fatalities. The demonstrated approach can be potentially suitable for a predictive mathematical model as long as the variance and mean of its dependent variable can be quantified with the TPL model.

MATERIALS AND METHODS

Taylor's Power Law

Compared with other PLs, TPL has two somewhat unique characteristics, both of which are determined by the two variables (variance and mean) it aims to quantify. The first is that

Abbreviations: AGM, American Gut Microbiome; DAR, Diversity-area relationship; DTR, Diversity-time relationship; FTR, Fatality-time relationship; MAD, Maximal accrual diversity; PLEC, Power law with exponential cutoff; SAR, Species-area relationship; STR, Species-time relationship; TPL, Taylor's power law.



(A) Coupling TPL and PLEC-FTR for COVID-19 Fatality Prediction

Continents	T_1	T_2	……	T_J	*Mean and Variance of Fatalities*
C_1	F_{11}	F_{12}	……	F_{1J}	$\overline{F_1} = \frac{1}{J}\sum_{j=1}^{J} F_{1j} \quad V_1 = \frac{1}{J}\sum_{j=1}^{J} (\overline{F_1} - F_{1j})^2$
C_2	F_{21}	F_{22}	……	F_{2J}	$\overline{F_2} = \frac{1}{J}\sum_{j=1}^{J} F_{2j} \quad V_2 = \frac{1}{J}\sum_{j=1}^{J} (\overline{F_2} - F_{2j})^2$
……	……	……	……	……	……
C_I	F_{I1}	F_{I2}	……	F_{IJ}	$\overline{F_I} = \frac{1}{J}\sum_{j=1}^{J} F_{Ij} \quad V_I = \frac{1}{J}\sum_{j=1}^{J} (\overline{F_I} - F_{Ij})^2$

Taylor's Power Law

$$V = a\overline{F}^b$$
$$\ln(V) = \ln(a) + b\ln(\overline{F})$$

PLEC-FTR model

$$F = cT^w \exp(dT),$$
$$T_{\max} = -w/d$$
$$F_{\max} = c(-\frac{w}{d})^w \exp(-w) = cT_{\max}^w \exp(-w)$$

Confidence Interval for PLEC Prediction

$$lower = F - 1.96 \times \sqrt{V/n} \qquad lower = F_{\max} - 1.96 \times \sqrt{V_{\max}/n}$$
$$upper = F + 1.96 \times \sqrt{V/n} \qquad upper = F_{\max} + 1.96 \times \sqrt{V_{\max}/n}$$

(B) Coupling TPL and PLEC-DAR for Species Richness Prediction

No. of Samples	**Cumulative Species Richness**	**Mean and Variance of Diversity**
1	S_1	$\overline{D_1} = \frac{1}{1}\sum_{i=1}^{1} S_i \quad V_1 = \frac{1}{1}\sum_{i=1}^{1} (\overline{D_1} - S_i)^2$
2	S_2	$\overline{D_2} = \frac{1}{2}\sum_{i=1}^{2} S_i \quad V_2 = \frac{1}{2}\sum_{i=1}^{2} (\overline{D_2} - S_i)^2$
……	……	……
N	S_N	$\overline{D_N} = \frac{1}{N}\sum_{i=1}^{N} S_i \quad V_N = \frac{1}{N}\sum_{i=1}^{N} (\overline{D_N} - S_i)^2$

Taylor's Power Law

$$V = a\overline{D}^b$$
$$\ln(V) = \ln(a) + b\ln(\overline{D})$$

PLEC-DAR model

$$D = cA^z \exp(dA),$$
$$A_{\max} = -z/d$$
$$D_{\max} = c(-\frac{z}{d})^z \exp(-z) = cA_{\max}^z \exp(-z)$$

Confidence Interval for PLEC Prediction

$$lower = D - 1.96 \times \sqrt{V/n} \qquad lower = D_{\max} - 1.96 \times \sqrt{V_{\max}/n}$$
$$upper = D + 1.96 \times \sqrt{V/n} \qquad upper = D_{\max} + 1.96 \times \sqrt{V_{\max}/n}$$

FIGURE 1 | A diagram illustrating the coupling of TPL and PLEC models: for predicting COVID-19 fatality [**(A)** the left block] and American gut microbiome diversity [**(B)** the right block]. The top pair of boxes in both case studies illustrates the format of input data, the middle boxes specify the power law models, and the bottom boxes list the formulae for computing the CIs.

its scaling parameter (exponent) that measures the population (community) spatial heterogeneity or temporal stability. This has to do with the fact that the variance (V) to mean (M) ratio (V/M) is a measure of the dispersion of data points (population abundances or counts), while dispersion, aggregation, and heterogeneity essentially characterize the same or similar system properties [6, 16]. For example, the TPL scaling parameter (b) can be used to measure heterogeneity at population, community, and landscape levels, respectively, depending on the level, the TPL model is constructed. The second characteristic of TPL is also related to the variance and mean: the relationship can be utilized for designing sampling schemes since the variance (level of variation or heterogeneity) determines the sampling efforts (sample sizes) necessary for estimating the population (species) abundances reliably {e.g., [12, 17]}. We take the advantages of TPL in this study to improve the quality of prediction/estimation because variance or SD is the foundation for computing CI of estimation.

Taylor's power law is one form of PLs, and it establishes the relationship between the variance and the mean of a random variable Y (e.g., population counts or abundances of biological populations) as a power function:

$$Var(Y) = V = aM^b \tag{1}$$

where V and M are the variance and mean of random variable Y; a and b are the parameters that can be estimated by fitting TPL to the variance-mean pairs of a series of spatial or temporal samples of populations. TPL can be fitted by a simple log-transformation {e.g., [5, 7]}, which generates:

$$\ln(V) = \ln(a) + b\ln(M) \tag{2}$$

Alternatively, non-linear optimization techniques, such as Marquardt's algorithm [29] or Simplex optimization [30], can be used to fit TPL directly (i.e., Eq. 1). However, log-transformed linear fitting (Eq. 2) may actually have an advantage from the perspective of scale-invariance as mentioned in the introduction section previously.

Ma [23] extended TPL to the community level by specifying Y as species abundance, M as the mean species abundance (size) *per* species in a community, and V is the corresponding variance. By regressing V-M across a series of communities (samples), one obtains type-I TPL extension (TPLE) for community spatial heterogeneity and type-II TPLE for community temporal stability. Similarly, there were type-III for mixed-species spatial heterogeneity and type-IV for mixed species temporal stability. The four TPLEs have the exactly same mathematical form as the original TPL [1] and [2], but the variables and parameters are defined and interpreted differently. Taylor [5] conjectured that TPL is only applied to integers, such as population counts (abundances), and it works poorly for ratios and very poorly for bounded ratios.

In this study, we take the advantages of TPL/TPLEs to estimate variance (V) corresponding to mean (M). The variance or its squared root (SD) provides necessary quantities for estimating CIs of PL or PLEC models as introduced below.

Power Law With Exponential Cutoff Model

Power law with exponential cutoff is a variant of PL model, and it was initially used to extend another classic PL in ecology, i.e., the SAR [31, 32]. The PL model for SAR is:

$$S = cA^{z} \tag{3}$$

where S is the number of species and A is the area of habitat occupied by S species.

Ma [25] extended the SAR to the general DAR by replacing the species richness (number of species) with general biodiversity (in Hill numbers).

$${}^{q}D = cA^{z} \tag{4}$$

where ${}^{q}D$ is diversity measured in the *q-th* order Hill numbers, A is the area, and c and z are parameters.

The PLEC model for DAR is:

$${}^{q}D = cA^{z}\exp(dA), \tag{5}$$

where d is a third parameter (taper-off parameter) and should be negative in DAR scaling models, and $\exp(dA)$ is the exponential decay term that eventually overwhelms the PL behavior at a very large value of A. The PLEC was originally introduced to SAR modeling by Plotkin et al. [33] and Ulrich and Buszko [34] (also see [35]), and Ma [25] extended it to DAR.

Ma [25] further derived the asymptote of the PLEC model and termed it as the MAD or potential diversity.

$$A_{\max} = -z/d \tag{6}$$

${}^{q}D$ may have a maximum in the following form:

$$Max({}^{q}D) = c\left(-\frac{z}{d}\right)^{z}\exp(-z) = cA_{\max}^{z}\exp(-z) \tag{7}$$

There are similar STR and corresponding DTR [27, 32]. STR/DTR has the exactly same PL/PLEC models as SAR/DAR described previously, but the data used to fit the models are different and so do the model parameters [27]. As further explained in the next sub-section, the fitting of PLEC can be performed with non-linear optimization, although log-transformed linear fitting, similar to fitting of TPL, can be used.

Ma [28] adapted the STR/DTR model to predict the inflection (turning) points of COVID-19, in which maximal accrual or potential diversity is equivalent to maximal infection numbers. In STR/DTR modeling, a convention is to use parameter w in place of the z of SAR/DAR as a diversity-time scaling parameter.

In the present study, we used the PLEC-DAR model to demonstrate the prediction of gut microbiome diversity and the PLEC-DTR model to demonstrate the prediction of COVID-19 fatalities, both augmented by the TPL model to get their CIs, as outlined below:

Coupling TPL and PLEC Models for Predicting the Interval of COVID-19 Fatalities

Here, we outline the integration of TPL with PLEC for predicting the interval of COVID-19 fatalities as following steps (also see **Figure 1**).

Step (*i*) Use the PLEC model (Eq. 5), adapted for fitting the fatality-time relationship (FTR) datasets as follows, i.e.,

$$F = cT^{w}\exp(dT), \tag{8}$$

where T is the time in days, and F is the fatality, c, w, and d are PLEC-FTR parameters. When the taper-off effects of parameter d is usually rather weak before the fatality numbers reach the peak, it is reasonable to treat w as an approximation to the *fatality growth rate* and c as an approximation to the *initial fatality number*. To fit PLEC-FTR model (Eq. 8), we adopted a non-linear optimization algorithm implemented as an R function "nlsLM" in R package "minpack.lm" (https://www.rdocumentation.org/packages/minpack.lm/versions/1.2-1/topics/nlsLM) [36]. Since $T_{\max} > 0$ is a necessary condition for the PLEC model to be biomedically sound, a constraint $d < 0$ was imposed for the non-linear fitting of the PLEC-FTR model.

Step (*ii*) Compute maximal accrual fatality (MAF) number using eqns. [6] and [7], adapted as:

$$F_{\max} = c\left(-\frac{w}{d}\right)^{w}\exp(-w) = cT_{\max}^{w}\exp(-w) \tag{9}$$

$$T_{\max} = -w/d \qquad (w > 0, d < 0) \tag{10}$$

Step (*iii*) Use TPL model (Eq. 1) for fitting the spatiotemporal aggregation (heterogeneity) of fatality numbers, i.e., adapting the original TPL (Eq. 1) as the following TPL for fatality aggregation:

$$V = a\bar{F}^{b} \tag{11}$$

where $\bar{F}$ is the mean fatality number of COVID-19 and V is the corresponding variance; a and b are the parameters. Parameters a and b are estimated by fitting Eq. [11] to spatiotemporal data of COVID-19 fatality, using the same scheme/procedures as used for fitting TPL to COVID-19 infection numbers [28].

Step (*iv*) Compute the variance (V) and SD ($\sqrt{V}$) based on Eq. [11] for fatality (F) (Eq. 8) or MAF ($F_{\max}$) (eqn. 9).

Step (*v*) Compute the lower and upper limits of 95% CI of COVID-19 fatality with the following pair of equations:

$$lower = F - 1.96\times\sqrt{V/n} \tag{12a}$$

$$lower = F_{\max} - 1.96\times\sqrt{V_{\max}/n} \tag{12b}$$

$$upper = F + 1.96\times\sqrt{V/n} \tag{13a}$$

$$upper = F_{\max} + 1.96\times\sqrt{V_{\max}/n} \tag{13b}$$

where n is the number of time points that correspond to F or F_{max} in (eqns. 8 and 9).

With eqns. (12a) and (13a), one can obtain the CI of COVID-19 fatalities at any time (day) points; alternatively, with eqns.

(12b) and (13b), one can obtain the CI of maximal accrual of COVID-19 fatality.

When $F_{\max}$ cannot be predicted (too early to predict), the PL model for FTR can be used to complete the above procedures for estimating the intervals of *F,* i.e., by setting $d = 0$, there is a PL model for $F = cT^w \exp(dT) = cT^w$.

Coupling TPL and PLEC Models for Predicting the Gut Microbiome Diversity

Similar to the previous integration of TPL and PLEC for estimating the CIs of COVID-19 fatalities, here we specify the procedures for predicting the Cis of AGM diversity (also see **Figure 1**).

Step (*i*) Use PLEC model (Eq. 5) for fitting the DAR datasets, i.e.,

$$^qD = cA^z \exp(dA), \tag{14}$$

where A is the number of individuals, and ^qD is the AGM diversity in Hill numbers, *c, z,* and *d* are PLEC-DAR parameters. To fit the PLEC-DAR model, we use the same non-linear optimization procedures as described previously for COVID-19 fatality prediction.

Step (*ii*) Compute MAD number using eqns. [6] and [7].

Step (*iii*) Adapt the TPL model (Eq. 1) for fitting the mean diversity and variance relationship:

$$V = a\bar{D}^b \tag{15}$$

where $\bar{D}$ is the mean diversity (Hill numbers) of AGM and V is the corresponding variance; a and b are the parameters. Parameters a and b are estimated by fitting Eq. [15] to AGM diversity data, using the same scheme/procedures as described above for COVID-19 fatality prediction.

Step (*iv*) Compute the variance (V) and SD ($\sqrt{V}$) based on Eq. [15] for diversity (D) (Eq. 5) or MAD ($D_{\max}$) (eqn. 7).

Step (*v*) Compute the lower and upper limits of 95% CI of diversity with the following pair of equations:

$$lower = D - 1.96 \times \sqrt{V/n} \tag{16a}$$

$$lower = D_{\max} - 1.96 \times \sqrt{V_{\max}/n} \tag{16b}$$

$$upper = D + 1.96 \times \sqrt{V/n} \tag{17a}$$

$$upper = D_{\max} + 1.96 \times \sqrt{V_{\max}/n} \tag{17b}$$

where n is the number of samples corresponding to D (Eq. 5) or $D_{\max}$ (Eq. 7). With eqns. (16a) and (17a), one can obtain the CI of diversity at any diversity accrual points; alternatively, with eqns. (16b) and (17b), one can obtain the CI of maximal accrual of diversity in Hill numbers.

When $D_{\max}$ cannot be predicted (too early to predict), the PL model for DAR can be used to complete the above procedures for estimating the intervals of D, i.e., by setting $d = 0$, there is a PL model for $D = cA^z \exp(dA) = cA^z$.

RESULTS

Coupling TPL and PLEC-FTR for Predicting the Intervals of COVID-19 Fatalities

The worldwide COVID-19 fatality numbers are available from the following website (https://github.com/CSSEGISandData/COVID-19) managed by Johns Hopkins University. Since the objective of this study was to demonstrate the feasibility of the coupling PL approach, we only extracted continent-level data for demonstrative purposes. For the country-level predictions, which are too extensive to cover in this article, we have another separate report.

Figure 1A illustrates the procedures to predict COVID-19 fatality, and **Table 1** lists the predictions for six continents and the whole world. The PLEC modeling succeeded in all continents and the world, except for Asia. The failure in Asia should be that the new wave of the outbreak in India was still too early to foresee the turning point of fatality, as discussed in Ma [28] for the similar prediction of COVID-19 infections.

In **Table 1**, the first five columns are self-evident given they are simply the PLEC-FTR parameters. The next three columns are the predictions by the PLEC model, the MAF (number) (F_{max}), and the days (T_{max}) (Julian days or Calendar date) at which F_{max} occurs. The next column is the actual fatality numbers at May 21, 2021, which happened to be the date we had completed the modeling work of this study, and which was listed to allow for a quick and rough reality check. The next column is the "completion level"—the percentage of past fatality over MAF (F_{max}). The last two columns are the novel contribution of this study, i.e., the lower and upper limits of predicted fatality numbers, which are not possible without the coupling of both the PLs (TPL and PLEC-FTR models).

Table 2 lists the fatality prediction for Asia based on the PL-FTR model, for which the PLEC model was failed. The predictions of the PL model should be treated with caution and are only of a rough reference value. As explained previously, when the PLEC-FTR modeling efforts fail, it is usually that the outbreak is still in early stage and there are not yet sufficiently long time-series datasets to allow for the fitting of the PLEC model. Although the PL-FTR model can be fitted in these cases, the predictions from the PL model are not sufficiently reliable.

Similar to the predictions of COVID-19 infections [28], there are some standard pre-processing procedures to take before fitting the PLEC-FTR to the fatality-time (day) datasets. For example, proper selection of starting point by truncating early data points (possibly including whole previous pandemic waves) could be necessary for successful model fitting. In fact, the fitting results presented in **Table 1** are obtained by setting the starting date for modeling on March 21, 2021 (until May 21, 2021). As discussed in detail by Ma [28], the selection of starting points does not influence the correctness of prediction since the infection (or death) numbers before truncation points are accumulated and treated as new starting infection (fatality) numbers for model-building.

Figure 2 displays the fitting of the TPL model to the COVID-19 fatality datasets, and the TPL parameters are used to compute the CIs for the fatality number prediction from the PLEC-FTR

TABLE 1 | The power law with exponential cutoff for fatality-time relationship (PLEC-FTR) model fitted with nonlinear optimization for daily cumulative counts of COVID-19 fatality, augmented with Taylor's power law (TPL) to obtain the 95% CIs*.

Continent	z	d	C	R^2	T_{max}	T_{max} (Date)	F_{max}	Observed (May 21, 2021)	Completion level (%)	Lower limit (95%)	Upper limit (95%)
Africa	1.150	−0.002	180.452	1.000	501	3-Aug-2022	182,643	127,983	70.1	169,865	195,420
Asia**	1.876	0.000	97.019	0.999	NA	NA	NA	636,068	NA	NA	NA
Europe	1.301	−0.012	1,734.846	1.000	113	11-Jul-2021	1,100,080	1,060,982	96.5	929,517	1,270,643
North America	1.185	−0.009	983.515	0.999	129	28-Jul-2021	875,359	854,545	97.6	749,159	1,001,560
South America	1.323	−0.007	1,504.372	1.000	193	29-Sep-2021	952,175	762,185	80.0	839,676	1,064,675
Oceania	1.413	−0.007	0.514	0.989	197	1-Oct-2021	1,191	1,095	92.0	1,075	1,306
World#	1.248	−0.003	4,957.140	1.000	485	19-Jul-2022	5,917,523	3,442,873	58.2	5,452,899	6,382,148

** Using fatality-time (date) relationship data from March 21 to May 21, 2021.*

*** PLEC failed for the dataset of Asia and PL model was fitted to Asia dataset successfully (see **Table 2**).*

TABLE 2 | The power law for fatality-time relationship (PL-FTR) model fitted for the daily cumulative counts of COVID-19 fatality, augmented with Taylor's power law (TPL) to obtain the 95% CIs.

Continent	z	ln(c)	R	P-value	Observed (May 21, 2021)	Predicted (May 21, 2021)	Predicted (June 21, 2021)	Predicted (July 21, 2021)	Predicted (Aug 21, 2021)	Start date
Asia	2.072	0.498	0.994	0.000	636,068	606,878 [562,269;651,487]	687,070 [637,883; 736,257]	772,420 [718,484; 826,356]	862,949 [804,096; 921,802]	10-Feb-2020

model. **Figure 3** displays the predicted COVID-19 fatalities based on the results, which are listed in **Table 1**.

Coupling TPL and PLEC-DAR for Predicting the Intervals of Gut Microbiome Diversity

Figure 1B shows the procedures for integrating the TPL and PLEC-DAR PL models for estimating the CIs of AGM diversity. The AGM datasets used to perform this demonstration are available for downloading in the public domain (http://americangut.org).

Table 3 exhibits the results from implementing the coupled TPL and PLEC-DAR modeling analysis. The first five columns in **Table 3** are simply the parameters of the fitted PLEC-DAR model for the AGM datasets, and the last four columns are simply the predicted MAD (species richness) of the AGM, i.e., the maximal accrual species richness (D_{max}) and the lower and upper limits of D_{max}. A_{max} is the number of individuals (sample sizes) at which the D_{max} is reached. Given that the samples of 1,473 individuals are used to build the PLEC-DAR model, and the A_{max} implies that 533 ($=A_{max}-1{,}473$, where $A_{max} = 2{,}006$, see **Table 3**) additional individuals are required to accumulate the maximal accrual species richness in the AGM cohort or population. **Figure 4** illustrates the fitting of the TPL model, which helps the estimation of the 95%-level CIs of D_{max}. **Figure 5** illustrates the predicted species richness (D_{max}) (the solid curve in red color) and its CI (dashed lines) and the observed species richness (the solid dots in blue color).

Numerous mathematical models have been produced to forecast the future of COVID-19 epidemics, but models are not crystal balls for predictions [37]. In particular, estimates from models about COVID-19 can contribute to uncertainty and anxiety to the public, lowering uncertainty can be helpful for alleviating possible anxiety accordingly. Jewell et al. [37] argued that short-term prediction can be critical for assisting the planning, but it is usually less productive to focus on long-term "guesses" for such purposes. The demonstrated application of the coupling PLs can lower the uncertainty of fatality prediction, besides being particularly simple and effective for short term (e.g., one epidemic wave of a pandemic) forecasting.

There are many alternative models to our proposed approach. For example, Li et al's [38] editorial introduces a series of 34 articles, published in the journal "*Frontiers in Physics*", on COVID-19 predictive modeling covering models/methods from classic Susceptible, Infectious, and/or Recovered (SIR) model and the associated reproductive number of the SIR to Gaussian model for the time evolution of the first corona pandemic wave. The Gaussian model is arguably the simplest analytically tractable model that allows for quantitative prediction of the time evolution of infections and fatalities during a pandemic wave. It can be rather challenging to compare and evaluate specific models, although rigorously evaluations and validations of model predictions are critical for their applications. For this, we feel it is beyond the scope of this article to compare our method with the existing models, especially those for COVID-19 predictions. On the other hand, we would like to present a brief discussion on the general strategy for building mathematical models in the section "Conclusions and Discussion."

Before concluding this subsection, an interesting phenomenon regarding the applications of artificial intelligence

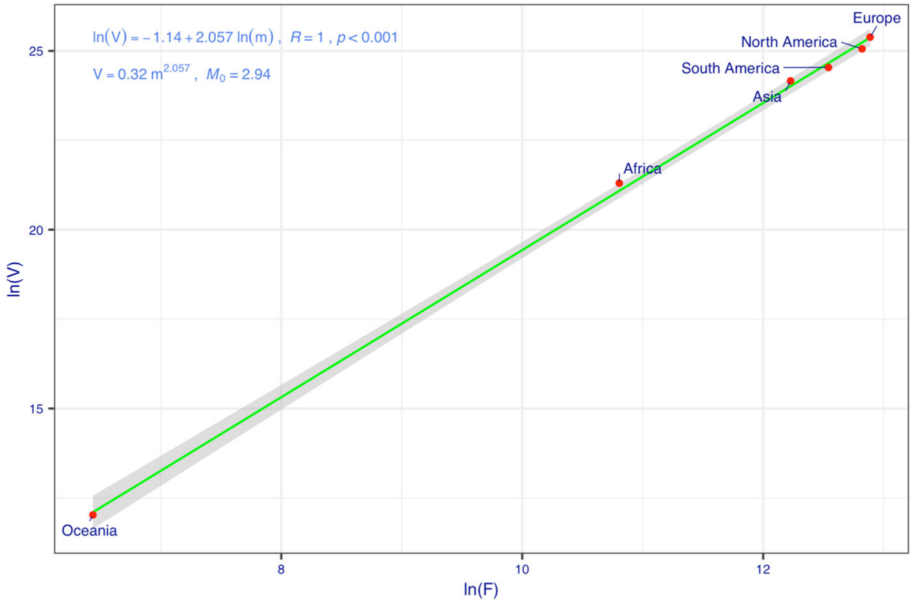


FIGURE 2 | Taylor's power law (TPL) model for the cumulative counts of COVID-19 fatalities: the variance corresponding to the mean fatality (F) is used to compute the SE and width of CI.

(AI) and machine learning (ML) to COVID-19 predictions seems to be worthy of particular notice. Vytla et al. [39] reviewed a slightly surprising phenomenon: the prediction of the COVID-19 pandemic is described as "the kryptonite of modern AI" and many predictions "by AI and ML are neither accurate nor reliable." The failure of AI can be due to an array of factors, and most prominent includes the lack of sufficient historic data to train AI models and the low quality of big data, often collected from social media. Even though the "garbage-in-garbage-out" is a well-known trap to modelers, the failure of AI models for COVID-19 predictions just reminds us that AI or ML is not an exception. In fact, the failure of big data in predicting epidemics occurred prior to the COVID-19 pandemic, for example, the failure of legendary Google Flu Trends (GFT) (https://www.wired.com/2015/10/can-learn-epic-failure-google-flu-trends/). According to Vytla et al. [39], the failure of AI and big data modeling has led to the enthusiasm to simple and traditional mathematical models for COVID-19 predictions. From this perspective, the simple PL approach we demonstrated in this study can be counted as another successful example. However, it should be emphasized that Vytla et al. [39] review and the previously discussed opinions on AI/ML may be limited to the predictions of epidemics/pandemics, and they can still be very useful for other problems of epidemics/pandemics.

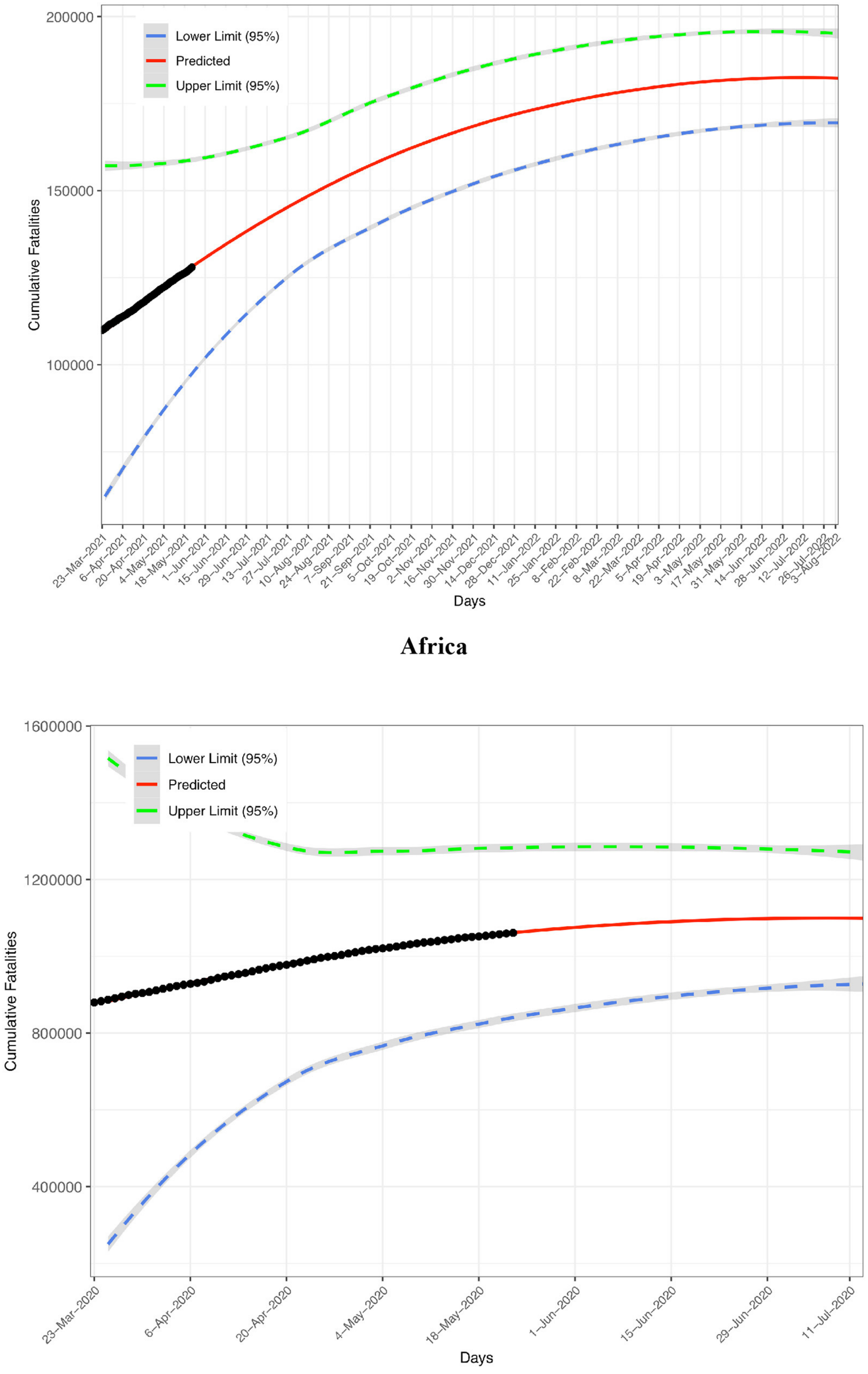
CONCLUSIONS AND DISCUSSION

The following findings can be summarized from previous sections:

(i) Coupling of TPL and PLEC models, the two PLs from classic ecological theories with applications beyond their original domain of ecology and biology, offers a feasible solution for some important prediction problems of power-law phenomena. We demonstrate the approach with two examples.

(ii) For the COVID-19 prediction problem, the PLEC-FTR model is able to predict the turning (inflection) points of fatality in the form of ($F_{\max}$, $T_{\max}$), i.e., the MAF number and corresponding date at which $F_{\max}$ is reached. In a previous study, we have demonstrated that the PLEC model successfully predicted the turning points of COVID-19 infections [28]. Both fatality and infection prediction problems are essentially the same, and therefore, prediction of fatality is undoubtedly feasible. An issue with our previous infection prediction is the lack of CI [28]. Thanks to the coupling with the TPL model, the PLEC-FTR is able to deliver the CI for $F_{\max}$ by leveraging the capability of TPL in predicting variance (SD) at different fatality levels. This is because the TPL in the case of fatality prediction can be harnessed to establish the power-function relationship between mean fatality number and corresponding variance. With the variance (SD), estimation of CIs is then a trivial statistical exercise. Obviously, the coupling approach is equally applicable to the prediction of COVID-19 infections, although it was not recognized [28]. This example also suggests that the TPL-PLEC coupling approach may be applied to other similar predictive problems in epidemiology and public health.

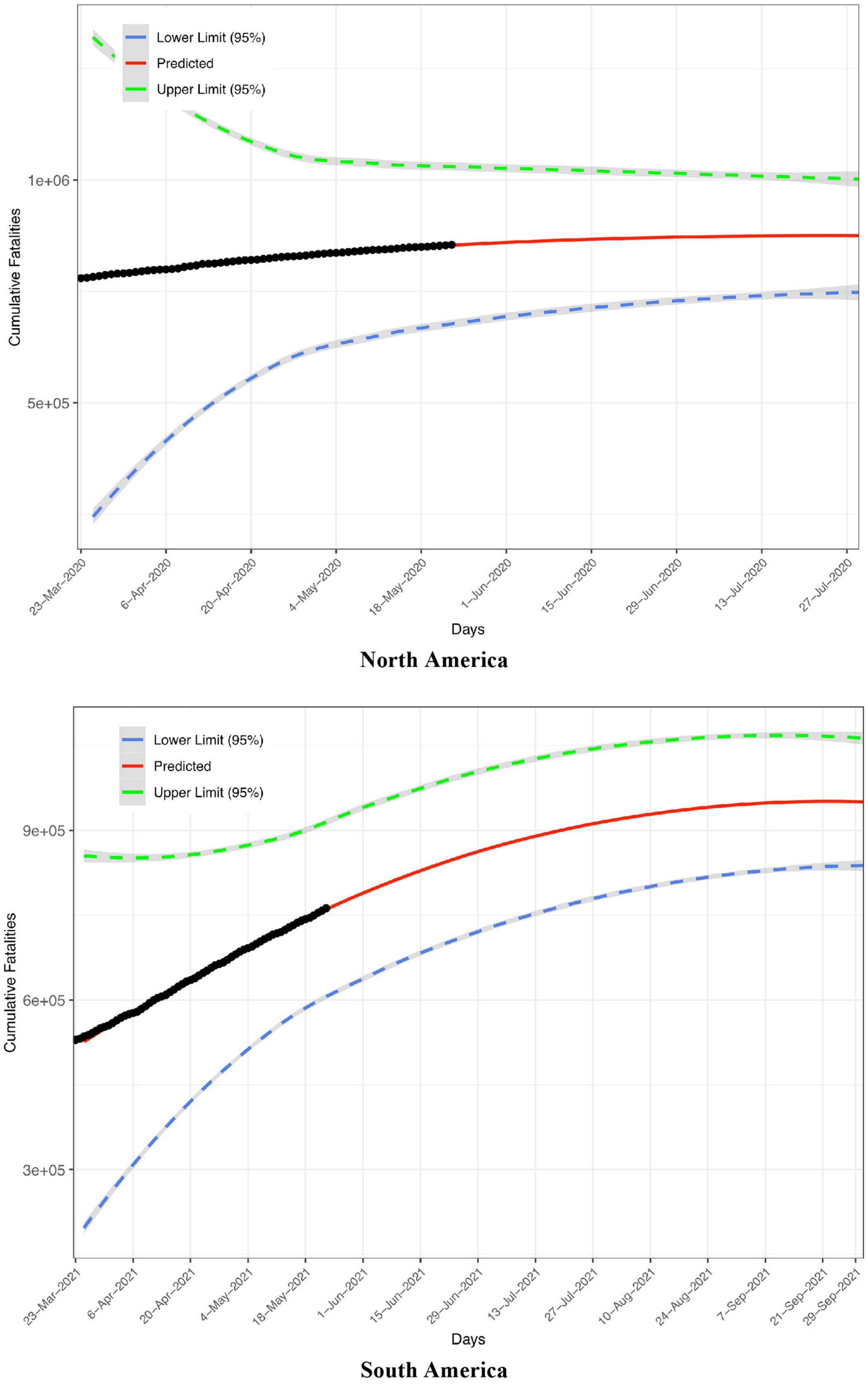
(iii) For the biodiversity prediction of AGM diversity, the coupling of TPL and PLEC-DAR models is able to predict the maximal accrual species richness ($D_{\max}$) of AGM, which can be considered as potential or "dark" species richness of gut microbiomes in the American cohort (population). The potential or dark biodiversity refers to the total diversity that



Africa

Europe

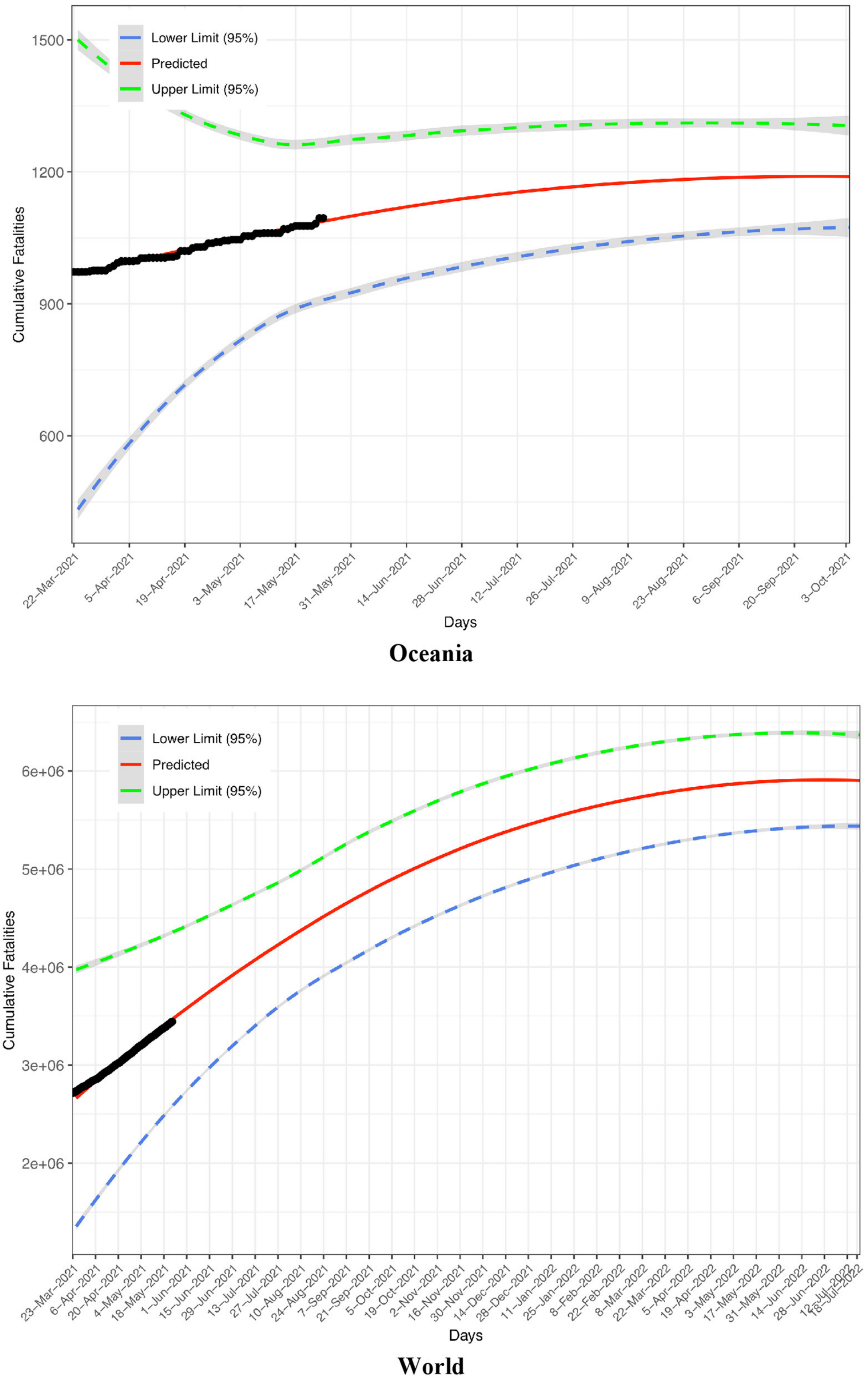
FIGURE 3 | Continued



North America

South America

FIGURE 3 | Continued



Oceania

World

FIGURE 3 | Predicted fatality number (solid curve in red), lower and upper bounds (dashed lines), and observed fatality number (solid cycles in black) for five continents and the world: Africa, Europe, North America, South America, Oceania, and the World.

TABLE 3 | The power law with exponential cutoff for diversity-area relationship (PLEC-DAR) model fitted with 1,000 times of re-sampling of the American gut microbiome (AGM) datasets consisting of 1,473 AGM samples, augmented with Taylor's power law (TPL) to obtain the 95% CIs.

Dataset	z	d	ln (c)	R	A_{max}	D_{max}	Lower limit (95%)	Upper limit (95%)
AGM Species Richness	0.386	−0.0002	6.598	0.995	2,006	9,414	9,310	9,518

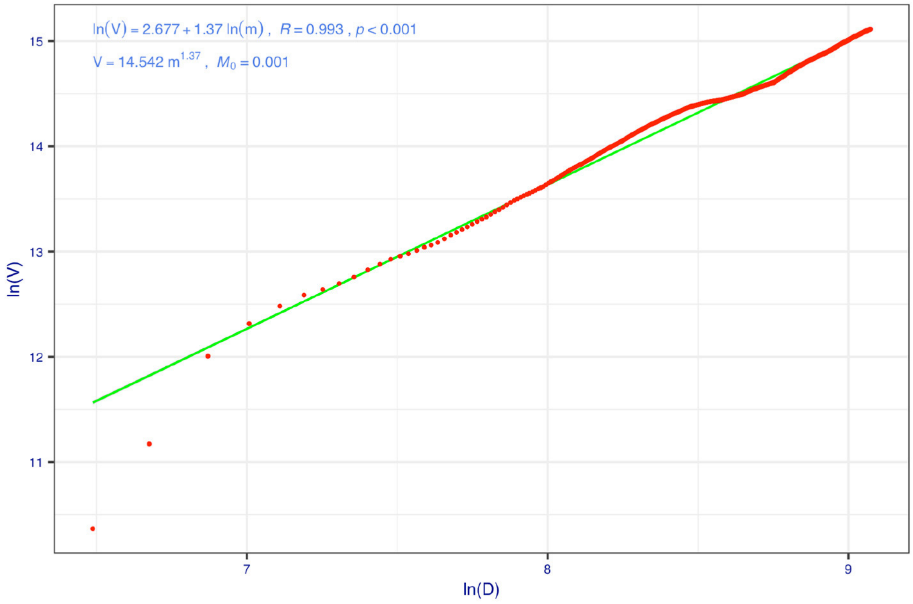


FIGURE 4 | Taylor's power law (TPL) model for the cumulative species richness of American gut microbiome (AGM) data set (the 100 times of re-sampling were used to fit 100 PL models, and here is one example): for each time of re-sampling, there are 1,473 pairs of variance/mean of species richness, computed for each step of diversity-area relationship (DAR) accumulation.

includes the portion that may be absent locally but is present in the regional species pool (and therefore is able to colonize local communities through dispersal/migration) [26]. In the case of the human gut microbiome, the potential diversity can be considered as a cohort or population level characteristic of the gut microbiome. In the case of this study, it can represent the potential species richness of the American population, given the datasets were obtained from sampling 1,473 Americans, a sufficiently large sample size.

In perspective, we expect that the power-law coupling approach possesses great promises for a wide range of important problems whenever both TPL and PLEC models can be successfully applied. The precondition that both PL models must be reliably built also reminds us that the approach cannot be a silver-bullet solution. For example, in the case of PLEC-DAR modeling for the gut microbiome diversity, we only presented the results for species richness (i.e., the Hill numbers when diversity order $q = 0$). The reason was that TPL was failed to fit the mean and variance of the Hill numbers at other diversity orders. This made it infeasible to estimate the CIs for other diversity orders. TPL has been found applicable in many natural and man-made systems; however, there are situations where it may fail. Taylor [5] conjectured that TPL might work poorly for ratios and very poorly for bounded ratios. The Hill number at diversity $q = 0$ (i.e., species richness) is an integer, but at other diversity orders, such as $q = 1, 2,$ or 3, the Hill numbers are indeed bounded ratios. Taylor's [5] conjecture may explain the limitation of TPL in fitting the mean-variance relationship in measuring biodiversity.

Furthermore, the *universality* property of PLs hints great promises for our coupling approach, although there have been occasional debates on proving universally in practical data fitted to PLs {e.g., [4]}. The universality refers to the equivalence of PLs with a particular scaling parameter (exponent), such as b in TPL, z in SAR (DAR), or w in STR (DTR), which are termed *critical exponents*. Critical exponents are termed so because the PL distributions of certain quantities are associated with phase transitions in dynamic systems as they approach criticality. The hallmark of universality is therefore the sharing of dynamics, and the systems with precisely the same critical exponents are said to belong to the same universality class. In the field of TPL, the transitions between aggregated (heterogeneous), random (Poisson), and uniform distribution of biological population or species abundance distribution can be characterized by the population aggregation critical density (PACD) [13] or

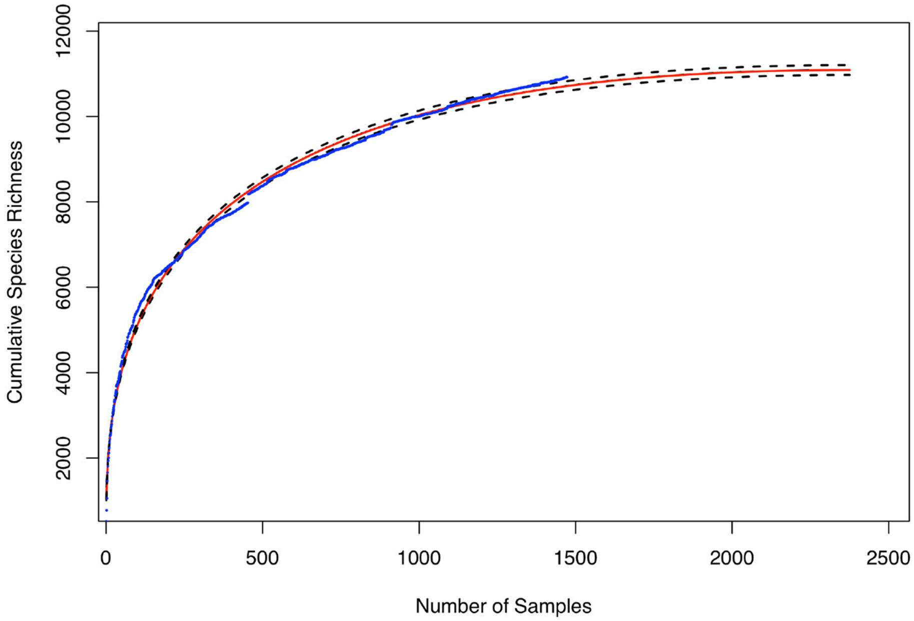


FIGURE 5 | Predicted species richness (solid curve in red color) of American gut microbiome (AGM) that includes lower and upper bounds (dashed lines) and observed species richness (solid tiny circles in blue color).

community heterogeneity critical diversity (CHCD) [23], which could be generated by self-organizations in the ecosystems (e.g., population or community). Different from physics, the processes, such as self-organization in biology and ecology, are difficult to prove rigorously. Nevertheless, there are indeed observations of the equivalence of TPL scaling exponents, such as the apparent invariance (constancy) of TPL scaling parameter (b) of global hot spring microbiomes across wide ranges of pH values and temperatures [40]. If these observations are found general in ecosystems, then the predictions based on our coupling approach of PLs can be not only feasible but also be reliable. Unlike the events that are governed by the normal (Gaussian) distribution, the events governed by the highly skewed PL distribution are particularly challenging to predict. In particular, some PL-governed events often lack of well-defined average values, but with potentially unbounded variance, tend to be black-swan and/or catastrophic. This also makes our proposed coupling method particularly valuable potentially.

Finally, we would like to present a very brief discussion on the general modeling strategy that is related to the two demonstrative case studies for illustrating the applications of the proposed coupling PLs. Since modeling strategy may be influenced by domain-specific knowledge, the discussion below is conducted in the context of ecological modeling {e.g., [41]} and COVID-19 prediction {*e.g.*, [28, 37]}, to keep relevant to the two demonstrative examples of this article. According to Levins [42], it is ideal to operate with manageable models that maximize generality, realism, and precision toward the overlapping, but not identical goals of understanding, predicting, and modifying nature. Levins [42] distinguished three alternative strategies, namely, [1] sacrifice generality to realism and precision (which is the approach of most simulation models); [2] sacrifice realism to generality and precision (most physicists who work in population ecology follow this tradition; the Lotka-Volterra model is an example); and [3] sacrifice precision to realism and generality, an example of this strategy is the theory of island biogeography by MacArthur and Wilson [43], which we have briefly discussed in the final paragraphs of this article. It is noted that the term "precision" here, more precisely, refers to more specific or detailed factors (information) used in modeling works.

Although Darwin's evolutionary theory answered the question of where and how biological species are originated and evolved on the earth's planet, the evolutionary theory did not explain how and why species co-exist and form diverse communities of species. Indeed, the competition or struggle for living, one essential aspect of evolutionary theory, would predict that the earth could be dominated by a handful of ultimate winners from competitions, which is obviously not consistent with the reality that the earth is cohabited by diverse species that usually coexist. In fact, biodiversity has been studied and paid attention by both scientists and the general public extensively in modern societies [44]. The study of biodiversity distribution, known as biogeography, was stuck in a "natural history phase" until the 1960s, due to the dominance by the collection of data and description of species, which were necessary but not sufficient. MacArthur and Wilson [43] demonstrated in their landmark monograph "*The Theory of Island Biogeography*" that the first principles of population ecology and genetics can be applied to explain how distance and area combine to regulate the balance between immigration and extinction in island populations. They were motivated to stimulate new forms of theoretical and empirical studies, rather than synthesizing and unifying existing theories or establishing a general new theory. Somewhat contrary to their unassuming start, their work does lead to a stronger theory of biodiversity. Today, even a half of a century has passed, the monograph continues to be inspiring and remains at the center of discussions about the geographic distributions of species in biodiversity research. Here are mentioned the above historical episodes for two reasons. First, MacArthur and Wilson's [43] island biogeography theory is well recognized as a landmark breakthrough in biogeography and community ecology. It can be considered as an extremely successful example of the modeling strategy of sacrificing precision (details) to realism and generality. Second, one of the key elements of their theory is the SAR PL, which is one of the PLs coupled in this study, i.e., the DAR extended by Ma [25–27]. Both factors should have contributed to the success of the biodiversity and COVID-19 predictions demonstrated in this study.

Besides the frequent infeasibility in simultaneously maximizing generality, realism, and precision of mathematical models, another commonly encountered dilemma for modelers is the complex vs. simple models. According to Jewell et al. [37], intuitively, simpler models may offer less valid predictions due to their limited capacity in capturing complex and unobserved human mixing patterns and other time-varying properties of infectious disease spread. However, complex models can be no more reliable than simple ones if they fail to capture key aspects of the problem. In addition, complex models may produce the illusion of realism and make it prone to omit crucial points. Furthermore, outputs of complex models are usually more sensitive to changes in parametric assumptions and/or the estimations of external disease or environmental factors, such as the lengths of latent/infectious periods due to mutation of a pathogen [37]. Of course, the disadvantages of complex models are not necessarily the advantages of simpler models. On the other hand, simpler models are usually inexpensive to construct and manage, and they may provide adequate solutions under certain circumstances. We hope that this work proposes and demonstrates a simple modeling approach for certain problems where PLs are applicable.

Finally, one may wonder how accurate the prediction of our coupling power laws is in forecasting the worldwide COVID-19 fatality. Compared with the worldwide COVID-19 fatality number on January 24th, 2022 (when this paper is formally accepted and online), the error rate of the prediction with our coupled power laws, made in the May 2021 (based on the fatality data then alone) is approximately 7% only (i.e., the precision level is 93%).

Specifically, we computed the worldwide fatality on Jan 24, 2022 with the following parameters and formula: $F = CT^w \exp(dT) + F_0$, where $C = 4957.140$, $w = 1.248$, $T = 308$, $d = -0.003$, $F_0 = 2716229$ (the fatality number at the starting date of the model-building, i.e., March 21st, 2021 in the case of the world model). We obtain $F = 5226117$, i.e., the predicted fatality number on January 24, 2022, and the prediction is based on the power law model established with the worldwide fatality numbers before May 21st, 2021 (**Table 1**, **Figure 3**). According to the publicly released COVID-19 fatality (https://github.com/CSSEGISandData/COVID-19), the actual worldwide fatality number is 5610729 on Janurary 24th, 2022. The precision of point estimation is then 92.6 or 93% approximately. Furthermore, the 95% confidence interval of the estimation can be computed with Eqns. (12, 13) and is [4713112, 5739122].Therefore, the point estimation of the worldwide COVID-19 fatality number on January 24th 2022 does fall within the confidence interval with a precision level of 92.6%. In fact, these results (including **Table 1** and **Figure 3**) had already been released on May 23rd 2021 in the preprint of this article Ma [45].

Our model (**Table 1**, **Figure 3**) also predicted that the turning point (inflection point) of worldwide COVID-19 fatality would not occur until the July of 2022, which contrasts with the recent prediction made by Murray [46] who suggested that the "end of the pandemic is near" by March 2022.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. Available publicly: https://github.com/CSSEGISandData/COVID-19.

AUTHOR CONTRIBUTIONS

The author confirms being the sole contributor of this work and has approved it for publication.

FUNDING

This study received funding from the following sources: A National Natural Science Foundation (NSFC) grant (no. 31970116), Cloud-Ridge Industry Technology Leader Grant.

ACKNOWLEDGMENTS

The author appreciates the data curation and computational assistance from Dr. Li LW, the Chinese Academy of Sciences. The author is also deeply indebted to two expert reviewers for their meticulous review of and insightful comments and suggestions for my manuscript, which offered significant help to improve this work.

REFERENCES

1. Eisler Z, Bartos I, Kertesz J. Fluctuation scaling in complex systems: Taylor's law and beyond. *Adv Phys.* (2008) 57:89–142. doi: 10.1080/00018730801893043
2. Fronczak A, Fronczak P. Origins of Taylor's power law for fluctuation scaling in complex systems. *Phys Rev E.* (2010) 81:066112. doi: 10.1103/PhysRevE.81.066112
3. Eliazar I. Power laws*:* a statistical trek. Springer. (2020). doi: 10.1007/978-3-030-33235-8

4. Stumpf MPH, Porter MA. Critical truths about power laws. *Science.* (2012) 335:665–6. doi: 10.1126/science.1216142
5. Taylor RAJ. Taylor's power law: order and pattern in nature. Academic Press, London. (2019) p. 657.
6. Ma ZS, Taylor RAJ. Human reproductive system microbiomes exhibited significantly different heterogeneity scaling with gut microbiome, but the intra-system scaling is invariant. *Oikos.* (2020) 129:903–11. doi: 10.1111/oik.07116
7. Taylor LR. Aggregation, variance and the mean. *Nature.* (1961) 189:732–5. doi: 10.1038/189732a0
8. Taylor LR, Taylor RAJ. Aggregation, migration and population mechanics. *Nature.* (1977) 265:415–21. doi: 10.1038/265415a0
9. Taylor LR, Taylor RAJ, Woiwod IP, Perry JN. Behavioral dynamics. *Nature.* (1983) 303:801–4. doi: 10.1038/303801a0
10. Taylor LR, Perry JN, Woiwod IP, Taylor RAJ. Specificity of the spatial power-law exponent in ecology and agriculture. *Nature.* (1988) 332:721–2. doi: 10.1038/332721a0
11. Taylor LR. Assessing and interpreting the spatial distributions of insect populations. *Annual Review of Entomology, vol.* (1984) 29:321–57. doi: 10.1146/annurev.en.29.010184.001541
12. Taylor RAJ. Spatial distribution, sampling efficiency and Taylor's power law. *Ecol Entomol.* (2018) 43:215–25. doi: 10.1111/een.12487
13. Ma ZS. Further interpreted Taylor's power law and population aggregation critical density. *Trans Ecol Soc China*. (1991) 1991:284–8.
14. Ma ZS. Chaotic populations in Genetic Algorithms. *Applied Soft Computing*. (2012) 12:2409–24 doi: 10.1016/j.asoc.2012.03.001
15. Ma ZS. Stochastic populations, power law and fitness aggregation in Genetic Algorithms. *Fundamenta Informaticae.* (2013) 122:173–206. doi: 10.3233/FI-2013-787
16. Ma ZS. Assessing and interpreting the metagenome heterogeneity with power law. *Front. Microbio.* (2020) 11:648. doi: 10.3389/fmicb.2020.00648
17. Ma ZS. Estimating the optimum coverage and quality of amplicon sequencing with Taylor's power law extensions. *Front Bioeng Biotechnol*. (2020) 8:372. doi: 10.3389/fbioe.2020.00372
18. Oh J, Byrd AL, Park M, NISC Comparative Sequencing Program, Kong HH, Segre JA. Temporal stability of the human skin microbiome. *Cell.* (2016) 165:854–66. doi: 10.1016/j.cell.2016.04.008
19. Cohen JE. Statistics of primes (and probably twin primes) satisfy Taylor's Law from ecology. *Am Stat.* (2016) 70:399–404. doi: 10.1080/00031305.2016.1173591
20. Cohen JE, Schuster WSF. Allometric scaling of population variance with mean body size is predicted from Taylor's law and density-mass allometry. *Proc Natl Acad Sci U.S.A.* (2012)109:15829. doi: 10.1073/pnas.1212883109
21. Cohen JE, Xu M. Random sampling of skewed distributions implies Taylor's power law of fluctuation scaling. *Proc Natl Acad Sci U.S.A.* (2015) 112:7749. doi: 10.1073/pnas.1503824112
22. Reuman DC, Zhao L, Sheppard LW, Reid PC, Cohen JE. Synchrony affects Taylor's law in theory and data. *Proc Natl Acad Sci U.S.A.* (2017) 114:6788. doi: 10.1073/pnas.1703593114
23. Ma ZS. Power law analysis of the human microbiome. *Molecular Ecology*. (2015) 24:5428–45. doi: 10.1111/mec.13394
24. Ma ZS. Spatial heterogeneity analysis of the human virome with Taylor's power law. *Comput Struct Biotechnol J.* (2021) 19:2921–7 doi: 10.1016/j.csbj.2021.04.069
25. Ma ZS. Extending species-area relationships (SAR) to diversity-area relationships (DAR). *Ecol Evol.* (2018) 8:10023–38. doi: 10.1002/ece3.4425
26. Ma ZS. A new DTAR (diversity–time–area relationship) model demonstrated with the indoor microbiome. *J Biogeography.* (2019) 46. doi: 10.1111/jbi.13636
27. Ma ZS. Diversity time-period and diversity-time-area relationships exemplified by the human microbiome. *Sci Rep.* (2018) 8:7214. doi: 10.1038/s41598-018-24881-3
28. Ma ZS. Predicting the outbreak risks and inflection points of COVID-19 pandemic with classic ecological theories. *Advanced Science*. (2020) 7:2001530. doi: 10.1002/advs.202001530
29. Ma ZS. Fitting Taylor's power law curve by marquardt's optimum algorithm. *J Beijing Forestry University*. (1990) 12:6.
30. Ma ZS. Optimization of nonlinear ecological models with the accelerated simplex algorithm. *J Biomathematics*. (1992) 7:160–7.
31. Watson HC. *Remarks on geographic distribution of British plants*. London. (1835).
32. Preston FW. *Time and space and the variation of species*. *Ecology*. (1960) 41:611–27. doi: 10.2307/1931793
33. Plotkin JB, Potts MD, Yu DW, Bunyavejchewin S, Condit R, et al. Predicting species diversity in tropical forests. *Proc Natl Acad Sci U.S.A.* (2000) 97:10850–4. doi: 10.1073/pnas.97.20.10850
34. Ulrich W, Buszko J. Self-similarity and the species–area relation of Polish butterflies. *Basic Appl Ecol.* (2003) 4:263–70. doi: 10.1078/1439-1791-00139
35. Tjørve E. Shapes and functions of species–area curves (II): a review of new models and parameterizations. *J Biogeogr*. (2009) 36:1435–45. doi: 10.1111/j.1365-2699.2009.02101.x
36. Bates DM, Watts DG. *Nonlinear Regression Analysis and Its Applications*. NY, USA: Wiley. (2008).
37. Jewell NP, Lewnard JA, Jewell BL. Opinion: predictive mathematical models of the COVID-19 pandemic: underlying principles and value of projections. *JAMA*. (2020) 323:1893–4. doi: 10.1001/jama.2020.6585
38. Li HJ, Wang L, Wang Z, Du ZW, Xia C, Moustakas A, Pei S. Editorial: mathematical modelling of the pandemic of 2019 novel coronavirus (COVID-19): Patterns, dynamics, prediction, and control. *Front Phy*. (2021). doi: 10.3389/fphy.2021.738602
39. Vytla V, Ramakuri SK, Peddi A, Srinivas K, Ragav NN. Mathematical models for predicting covid-19 pandemic: A review. *J Phys Conf Ser*. (2021) 1797. doi: 10.1088/1742-6596/1797/1/012009
40. Li LW, Ma ZS. Comparative power law analysis for the spatial heterogeneity scaling of the hot-spring and human microbiomes. *Mol Ecol*. (2018) 28:2932–43. doi: 10.1111/mec.15124
41. Kingsland SE. *Modeling* Nature, Episodes in the History of Population Ecology. The University of Chicago Press. (1985).
42. Levins R. The strategy of model building in population biology. *Am Sci.* (1966) 54:421–31.
43. MarArthur RH, Wilson EO. The theory of island biogeography. *Princeton Landmarks in Biology.* (1967).
44. Ma ZS, Ellison AM. Dominance network analysis provides a new framework for studying the diversity-stability relationship. *Ecol Monographs*. (2019) 89. doi: 10.1002/ecm.1358
45. Ma ZS. Coupling power laws offers a powerful method for problems such as biodiversity and COVID-19 fatality predictions. *Quantitative Methods*. (2021). doi: 10.48550/arXiv.2105.11002
46. Murray CJL. COVID-19 will continue but the end of the pandemic is near. *The Lancet*. (2022). 399:417–9. doi: 10.1016/S0140-6736(22)00100-3

Conflict of Interest: The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 20 April 2022
doi: 10.3389/fams.2022.880086

Generalized Kibria-Lukman Estimator: Method, Simulation, and Application

Issam Dawoud[1], *Mohamed R. Abonazel*[2]* and *Fuad A. Awwad*[3]

[1] *Department of Mathematics, Al-Aqsa University, Gaza, Palestine,* [2] *Department of Applied Statistics and Econometrics, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza, Egypt,* [3] *Department of Quantitative Analysis, College of Business Administration, King Saud University, Riyadh, Saudi Arabia*

In the linear regression model, the multicollinearity effects on the ordinary least squares (OLS) estimator performance make it inefficient. To solve this, several estimators are given. The Kibria-Lukman (KL) estimator is a recent estimator that has been proposed to solve the multicollinearity problem. In this paper, a generalized version of the KL estimator is proposed, along with the optimal biasing parameter of our proposed estimator derived by minimizing the scalar mean squared error. Theoretically, the performance of the proposed estimator is compared with the OLS, the generalized ridge, the generalized Liu, and the KL estimators by the matrix mean squared error. Furthermore, a simulation study and the numerical example were performed for comparing the performance of the proposed estimator with the OLS and the KL estimators. The results indicate that the proposed estimator is better than other estimators, especially in cases where the standard deviation of the errors was large and when the correlation between the explanatory variables is very high.

Keywords: generalized liu estimator, multicollinearity, generalized ridge estimator, biasing parameter, ridge-type estimator

OPEN ACCESS

Edited by:
Min Wang,
University of Texas at San Antonio,
United States

Reviewed by:
Muhammad Suhail,
University of Agriculture,
Peshawar, Pakistan
Zakariya Yahya Algamal,
University of Mosul, Iraq

****Correspondence:***
Mohamed R. Abonazel
mabonazel@cu.edu.eg

Specialty section:
This article was submitted to
Statistics and Probability,
a section of the journal
Frontiers in Applied Mathematics and
Statistics

Received: *21 February 2022*
Accepted: *14 March 2022*
Published: *20 April 2022*

Citation:
Dawoud I, Abonazel MR and Awwad FA (2022) Generalized Kibria-Lukman Estimator: Method, Simulation, and Application. Front. Appl. Math. Stat. 8:880086. doi: 10.3389/fams.2022.880086

INTRODUCTION

The statistical consequences of multicollinearity are well-known in statistics for a linear regression model. Multicollinearity is known as the approximately linear dependency among the columns of the matrix X in the following linear model

$$y = X\beta + \varepsilon, \varepsilon \sim N\left(0, \sigma^2 I_n\right) \tag{1}$$

where y is an $n \times 1$ vector of the given dependent variable, X is a known $n \times p$ matrix of the given explanatory variables, β is an $p \times 1$ vector of given unknown regression parameters, and ε is described as an $n \times 1$ vector of the disturbances. Then, the ordinary least squares (OLS) estimator of β for the model (1) is given as:

$$\hat{\beta} = (X'X)^{-1}X'y$$

The multicollinearity problem effects on the behavior of the OLS estimator make it inefficient. Sometimes, it produces wrong signs [1, 2]. Many studies were conducted to handle this. For example, Hoerl and Kennard [2] proposed the ordinary ridge and the generalized ridge (GR) estimators, while Liu [3] introduced the popular Liu and the generalized Liu (GL), and very recently, Kibria and Lukman [1] proposed a ridge-type estimator called the Kibria–Lukman (KL) estimator which is defined by

$$\hat{\beta}_{KL} = (X'X + kI_p)^{-1}(X'X - kI_p)\hat{\beta}, k > 0$$

This estimator has been extended for use in different generalized linear models, such as Lukman et al. [4, 5], Akram et al. [6], and Abonazel et al. [7].

According to recent papers [8–10], we can say that the efficiency of any bias estimator will increase if the estimator is modified or generalized using bias parameters that vary from observation to observation in the sample (k_i and/or d_i) rather than in fixed bias parameters (k and/or d). Hence, the main purpose of this paper is to develop a general form of the KL estimator to combat the multicollinearity in the linear regression model.

The rest of the discussion in this paper is structured as follows: Section Statistical Methodology presents the statistical methodology. In Section Superiority of the Proposed GKL Estimator, we theoretically compare the proposed general form of the KL estimator with each of the mentioned estimators. In Section The Biasing Parameter Estimator of the GKL Estimator, we give the estimation of the biasing parameter of the proposed estimator. Different scenarios of the Monte Carlo simulation are done in Section A Monte Carlo Simulation Study. A real data is used in Section Empirical Application. Finally, Section Conclusion presents some conclusions.

STATISTICAL METHODOLOGY

Canonical Form

The canonical form of the model in equation (1) is used as follows:

$$y = Z\alpha + \varepsilon \tag{2}$$

where $Z = XR, \alpha = R^{'}\beta$, and R is an orthogonal matrix such that $Z'Z = R'X'XR = G = diag(g_1, g_2, \ldots, g_p)$. Then, the OLS of α is as:

$$\hat{\alpha} = G^{-1}Z'y \tag{3}$$

and the matrix mean squared error (MMSE) is given as,

$$MMSE\left(\hat{\alpha}\right) = \sigma^2 G^{-1} \tag{4}$$

Ridge Regression Estimators

The OR and the GR of $\alpha$are, respectively, defined as follows [2]:

$$\hat{\alpha}_{OR} = W_1 G\hat{\alpha} \tag{5}$$
$$\hat{\alpha}_{GR} = W_2 G\hat{\alpha} \tag{6}$$

where $W_1 = [G + kI_p]^{-1}$, $k > 0$ and $W_2 = [G + K]^{-1}$, with $K = diag(k_1, k_2, ..., k_p)$, $k_i > 0$, and $i = 1, 2, ..., p$.

The MMSE of the OR and the GR are given respectively as:

$$MMSE(\hat{\alpha}_{OR}) = \sigma^2 W_1 G W_1{}' + (W_1 G - I_p)\alpha\alpha'(W_1 G - I_p)' \tag{7}$$
$$MMSE(\hat{\alpha}_{GR}) = \sigma^2 W_2 G W_2{}' + (W_2 G - I_p)\alpha\alpha'(W_2 G - I_p)' \tag{8}$$

Liu Regression Estimators

The Liu and the GL of $\alpha$are respectively defined as follows [3]:

$$\hat{\alpha}_{Liu} = F_1\hat{\alpha} \tag{9}$$
$$\hat{\alpha}_{GL} = F_2\hat{\alpha} \tag{10}$$

where

$$F_1 = [G + I_p]^{-1}[G + dI_p], 0 < d < 1 \text{ and } F_2 = [G + I_p]^{-1} [G + D], \text{with } D = diag(d_1, d_2, ..., d_p) \text{ and } 0 < d_i < 1.$$

The MMSE of the Liu and the GL are, respectively, given as:

$$MMSE(\hat{\alpha}_{Liu}) = \sigma^2 F_1 G^{-1} F_1{}' + (F_1 - I_p)\alpha\alpha'(F_1 - I_p)' \tag{11}$$
$$MMSE(\hat{\alpha}_{GL}) = \sigma^2 F_2 G^{-1} F_2{}' + (F_2 - I_p)\alpha\alpha'(F_2 - I_p)' \tag{12}$$

Kibria–Lukman Estimator

The KL estimator of $\alpha$is given as Kibria and Lukman [1]:

$$\hat{\alpha}_{KL} = W_1 M_1\hat{\alpha} \tag{13}$$

where $M_1 = [G - kI_p]$ and the MMSE of this estimator is given as:

$$MMSE(\hat{\alpha}_{KL}) = \sigma^2 W_1 M_1 G^{-1} M_1{}' W_1{}' + [W_1 M_1 - I_p]\alpha\alpha'[W_1 M_1 - I_p]' \tag{14}$$

The Proposed GKL Estimator

Now, by replacing W_1 with W_2 and M_1 with $M_2 = [G - K]$ in the KL estimator, we obtain the general form of the GKL estimator as follows:

$$\hat{\alpha}_{GKL} = W_2 M_2\hat{\alpha} \tag{15}$$

then, the MMSE of the proposed GKL estimator is computed by,

$$MMSE(\hat{\alpha}_{GKL}) = \sigma^2 W_2 M_2 G^{-1} M_2{}' W_2{}' + [W_2 M_2 - I_p] \alpha\alpha'[W_2 M_2 - I_p]' \tag{16}$$

SUPERIORITY OF THE PROPOSED GKL ESTIMATOR

In this section, we make a comparison of the proposed GKL estimator with each of OLS, GR, GL, and KL estimators. First, we offer some useful lemmas for our comparisons of estimators.

Lemma 1: Wang et al. [11]: Suppose M and N are $n \times n$ positive definite matrices, then $M > N$ if and only if (iff) $\lambda - 1_{\max}$, where $\lambda - 1_{\max}$ is the maximum eigenvalue of NM^{-1} matrix.

Lemma 2: Farebrother [12]: Let S be an $n \times n$ positive definite matrix. That is, $S > 0$ and α be some vector. Then, $S - \alpha\alpha^{'} > 0$ iff $\alpha' S^{-1}\alpha < 1$.

Lemma 3: Trenkler and Toutenburg [13]: Let $\alpha_i = U_i w$, $i = 1, 2$ be any two linear estimators of α. Suppose that $Q = Cov(\hat{\alpha}_1) - Cov(\hat{\alpha}_2) > 0$, where $Cov(\hat{\alpha}_i), i = 1, 2$ be the covariance matrix of $\hat{\alpha}_i$ and $b_i = Bias(\hat{\alpha}_i) = (U_i X - I)\alpha$. Then,

$$\Delta\left(\hat{\alpha}_1 - \hat{\alpha}_2\right) = MMSE\left(\hat{\alpha}_1\right) - MMSE\left(\hat{\alpha}_2\right) = \sigma^2 Q + b_1 b_1{}' - b_2 b_2{}' > 0 \tag{17}$$

iff $b_2'[\sigma^2 Q+b_1'b_1]^{-1}b_2 < 1$ where $MMSE(\hat{\alpha}_i) = Cov(\hat{\alpha}_i)+ b_i b_i'$

Theorem 1: $\hat{\alpha}_{GKL}$ is superior to $\hat{\alpha}$ iff

$$\alpha'[W_2M_2 - I_p]'[\sigma^2(G^{-1} - W_2M_2G^{-1}M_2'W_2')] [W_2M_2 - I_p]\alpha < 1 \qquad (18)$$

Proof: The covariance matrices difference is written as

$$\begin{aligned} Difference &= \sigma^2\left(G^{-1} - W_2M_2G^{-1}M_2'W_2'\right) \\ &= \sigma^2 diag\left\{\frac{1}{g_i} - \frac{(g_i-k_i)^2}{g_i(g_i+k_i)^2}\right\}_{i=1}^{p} \end{aligned} \qquad (19)$$

where $G^{-1} - W_2M_2G^{-1}M_2'W_2'$ becomes positive definite iff $(g_i + k_i)^2 - (g_i - k_i)^2 > 0$ or $(g_i + k_i) - (g_i - k_i) > 0$. It is clear that for $k_i > 0, i = 1, 2, ..., p$, $(g_i + k_i) - (g_i - k_i) = 2k_i > 0$. Therefore, this is done using Lemma 3.

Theorem 2: When $\lambda - 1_{\max}$, $\hat{\alpha}_{GKL}$ is superior to $\hat{\alpha}_{GR}$ iff

$$\alpha'[W_2M_2 - I_p]'[V_1 + (W_2G - I_p)\alpha\alpha'(W_2G - I_p)'] [W_2M_2 - I_p]\alpha < 1 \qquad (20)$$

$$\lambda - 1_{\max} \qquad (21)$$

where $V_1 = \sigma^2(W_2GW_2' - W_2M_2G^{-1}M_2'W_2')$, $N = W_2KG^{-1}KW_2'$, and $M = 2W_2KKW_2'$.

Proof:

$$\begin{aligned} V_1 &= \sigma^2\left(W_2GW_2' - W_2M_2G^{-1}M_2'W_2'\right) \\ &= \sigma^2\left(W_2GW_2' - W_2\left(G-K\right)G^{-1}\left(G-K\right)W_2'\right) \\ &= \sigma^2\left(2W_2KKW_2' - W_2KG^{-1}KW_2'\right) \\ &= \sigma^2(M-N) \end{aligned}$$

For $k_i > 0$, it is obvious that $M > 0$ and $N > 0$. Then, $M - N > 0$ iff $\lambda - 1_{\max}$, where $\lambda - 1_{\max}$ is the maximum eigenvalue of NM^{-1}. So, this is done by Lemma 1.

Theorem 3: $\hat{\alpha}_{GKL}$ is superior to $\hat{\alpha}_{GL}$ iff

$$\alpha'[W_2M_2 - I_p]'[V_2 + (F_2 - I_p)\alpha\alpha'(F_2 - I_p)'][W_2M_2 - I_p] \alpha < 1 \qquad (22)$$

where $V_2 = \sigma^2(F_2G^{-1}F_2' - W_2M_2G^{-1}M_2'W_2')$.

Proof: The covariance matrices difference is written as

$$\begin{aligned} V_2 &= \sigma^2\left(F_2G^{-1}F_2' - W_2M_2G^{-1}M_2'W_2'\right) \\ &= \sigma^2 diag\left\{\frac{(g_i+d_i)^2}{g_i(g_i+1)^2} - \frac{(g_i-k_i)^2}{g_i(g_i+k_i)^2}\right\}_{i=1}^{p} \end{aligned} \qquad (23)$$

where $F_2G^{-1}F_2' - W_2M_2G^{-1}M_2'W_2'$ becomes positive definite iff $(g_i + k_i)^2(g_i + d_i)^2 - (g_i - k_i)^2(g_i + 1)^2 > 0$ or $(g_i + k_i)(g_i + d_i) - (g_i - k_i)(g_i + 1) > 0$. So, if $k_i > 0$ and $0 < d_i < 1$, $(g_i+k_i)(g_i+d_i)-(g_i-k_i)(g_i+1) = k_i(2g_i+d_i+1)+g_i(d_i-1) > 0$. So, this is done by Lemma 3.

Theorem 4: $\hat{\alpha}_{GKL}$ is superior to $\hat{\alpha}_{KL}$ iff

$$\alpha'[W_2M_2 - I_p]'[V_3 + (W_1M_1 - I_p)\alpha\alpha'(W_1M_1 - I_p)'] [W_2M_2 - I_p]\alpha < 1 \qquad (24)$$

where $V_3 = \sigma^2(W_1M_1G^{-1}M_1'W_1' - W_2M_2G^{-\ 1}M_2'W_2')$.

Proof: The covariance matrices difference is written as

$$\begin{aligned} V_3 &= \sigma^2\left(W_1M_1G^{-1}M_1'W_1' - W_2M_2G^{-1}M_2'W_2'\right) \\ &= \sigma^2 diag\left\{\frac{(g_i-k)^2}{g_i(g_i+k)^2} - \frac{(g_i-k_i)^2}{g_i(g_i+k_i)^2}\right\}_{i=1}^{p} \end{aligned} \qquad (25)$$

where $W_1M_1G^{-1}M_1'W_1' - W_2M_2G^{-1}M_2'W_2'$ becomes positive definite iff $(g_i + k_i)^2(g_i - k)^2 - (g_i - k_i)^2(g_i + k)^2 > 0$ or $(g_i + k_i)(g_i - k) - (g_i - k_i)(g_i + k) > 0$. So, if $k_i > 0$ and $k_i > k$, $(g_i + k_i)(g_i - k) - (g_i - k_i)(g_i + k) = 2g_i(k_i - k) > 0$. So, this is done by Lemma 3.

THE BIASING PARAMETER ESTIMATOR OF THE GKL ESTIMATOR

The performance of any estimator depends on its biasing parameter. Therefore, the determination of the biasing parameter of an estimator is an important issue. Different studies analyzed this issue (e.g., [2, 3, 8–10, 14–24]).

Kibria and Lukman [1] proposed the biasing parameter estimator of the KL estimator as follows:

$$\hat{k} = \min\left\{\frac{\hat{\sigma}^2}{[(\hat{\sigma}^2/g_i) + 2\hat{\alpha}_i^2]}\right\}_{i=1}^{p} \qquad (26)$$

Here, we find the estimation of the optimal values of k_i for the proposed GKL estimator. The optimal values of k_i are obtained by minimizing

$$MMSE(\hat{\alpha}_{GKL}) = E[(\hat{\alpha}_{GKL} - \alpha)'\left(\hat{\alpha}_{GKL} - \alpha\right)],$$

$$m(k_1, k_2, ..., k_p) = tr(MMSE(\hat{\alpha}_{GKL}), \text{and}$$

$$m(k_1, k_2, ..., k_p) = \sigma^2\sum_{i=1}^{p}\frac{(g_i-k_i)^2}{g_i(g_i+k_i)^2} + \sum_{i=1}^{p}\frac{4k_i^2\alpha_i^2}{(g_i+k_i)^2} \qquad (27)$$

Differentiating $m(k_1, k_2, ..., k_p)$ with respect to k_i and setting $[\frac{\partial m(k_1,k_2,...,k_p)}{\partial k_i}] = 0$, the optimal values of k_i after replacing σ^2 and α_i^2 by their unbiased estimators become as follows:

$$\hat{k}_i = \frac{\hat{\sigma}^2}{((\hat{\sigma}^2/g_i) + 2\hat{\alpha}_i^2)}, i = 1, 2, ..., p \qquad (28)$$

A MONTE CARLO SIMULATION STUDY

The explanatory variables are generated as follows [25–27]:

$$x_{ji} = (1 - \rho^2)^{\frac{1}{2}}a_{ji} + \rho a_{jp}, j = 1, 2, ..., n, i = 1, 2, ..., p \qquad (29)$$

where a_{ji} are the independent pseudo-random numbers that have the standard normal distribution and ρ is known that the correlation between two given explanatory variables. The dependent variable y are given by:

$$y_j = \beta_1x_{j1} + \beta_2x_{j2} + \ldots + \beta_px_{jp} + \varepsilon_j, j = 1, 2, \ldots, n \qquad (30)$$

TABLE 1 | The factors' values of the simulation study.

Factor	Symbol	Levels
Sample size	n	50, 100, 150
Standard deviation	σ	1, 5, 10
Degree of correlation	ρ	0.8, 0.9, 0.99
Explanatory variables number	p	3, 7
Replicates number	MCN	5,000

TABLE 2 | Estimated mean squared error (EMSE) values of the estimators when $p = 3$.

n	σ	ρ	OLS	KL	GKL
50	1	0.8	0.1249	**0.1094**	0.1548
		0.9	0.2260	**0.1829**	0.2738
		0.99	2.0641	1.1439	**1.1208**
	5	0.8	3.1235	1.7550	**1.6052**
		0.9	5.6491	2.8600	**2.4774**
		0.99	51.6036	22.2378	**17.6275**
	10	0.8	12.4940	6.2898	**5.3865**
		0.9	22.5965	10.5775	**8.7621**
		0.99	206.4144	87.8850	**69.2762**
100	1	0.8	0.0605	**0.0557**	0.0701
		0.9	0.1107	**0.0964**	0.1373
		0.99	1.0308	**0.6454**	0.7558
	5	0.8	1.5118	**0.9306**	0.9509
		0.9	2.7663	1.5097	**1.4056**
		0.99	25.7697	11.3736	**8.9376**
	10	0.8	6.0471	3.1436	**2.7244**
		0.9	11.0651	5.2952	**4.3648**
		0.99	103.0788	44.4958	**34.4270**
150	1	0.8	0.0420	**0.0393**	0.0469
		0.9	0.0768	**0.0687**	0.0928
		0.99	0.7125	**0.4700**	0.6113
	5	0.8	1.0497	**0.6763**	0.7487
		0.9	1.9189	1.0893	**1.0826**
		0.99	17.8124	7.7631	**6.1352**
	10	0.8	4.1988	2.2214	**1.9830**
		0.9	7.6756	3.6905	**3.1029**
		0.99	71.2496	29.9827	**23.1604**

For each case, the smallest EMSE value is bolded.

TABLE 3 | EMSE values of the estimators when $p = 7$.

n	σ	ρ	OLS	KL	GKL
50	1	0.8	0.4143	**0.3129**	0.4302
		0.9	0.6792	**0.5399**	0.6831
		0.99	7.3867	3.9941	**3.0983**
	5	0.8	10.3568	5.5139	**4.1882**
		0.9	19.4796	10.0849	**7.4658**
		0.99	184.6673	92.8175	**66.6994**
	10	0.8	41.4272	21.1839	**15.5082**
		0.9	77.9186	39.4124	**28.5547**
		0.99	738.6690	370.3048	**265.3667**
100	1	0.8	0.1766	**0.1529**	0.2137
		0.9	0.3322	**0.2702**	0.3652
		0.99	3.1561	1.9888	**1.7020**
	5	0.8	4.4159	2.7275	**2.2455**
		0.9	8.3060	4.8911	**3.8358**
		0.99	78.9019	43.6091	**32.3890**
	10	0.8	17.6638	10.1544	**7.7808**
		0.9	33.2240	18.6747	**14.0582**
		0.99	315.6077	173.4003	**128.2151**
150	1	0.8	0.1105	**0.0992**	0.1341
		0.9	0.2081	**0.1773**	0.2504
		0.99	1.9769	1.3108	**1.2036**
	5	0.8	2.7632	1.7804	**1.5371**
		0.9	5.2014	3.1588	**2.5389**
		0.99	49.4224	27.3769	**20.2601**
	10	0.8	11.0529	6.4542	**4.9732**
		0.9	20.8054	11.8006	**8.8790**
		0.99	197.6896	108.306	**79.6545**

For each case, the smallest EMSE value is bolded.

TABLE 4 | Estimated coefficients and mean squared error (MSE) values of the estimators.

Estimator	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	MSE
OLS	2.1930	1.1533	0.7585	0.4863	0.0638
KL	2.1764	1.1572	0.7465	0.4888	0.0629
GKL	2.1653	1.1613	0.7312	0.4904	0.0620

where ε_j are the $i.i.dN(0, \sigma^2)$. The values of β are given such that $\beta'\beta = 1$ as discussed in Dawoud and Abonazel [28], Algamal and Abonazel [29], Abonazel et al. [7, 30], and Awwad et al. [31]. Also, all factors that used in the simulation are given in **Table 1**.

In order to see the performance of the OLS, KL, and the proposed GKL estimators with their biasing parameters estimators presented in Section Statistical Methodology, the estimated mean squared error (EMSE) are calculated for each replicate with different values of σ, ρ, n, and p using the following formula:

$$EMSE(\alpha^*) = \frac{1}{MCN} \sum_{l=1}^{MCN} (\alpha_l^* - \alpha)'(\alpha_l^* - \alpha) \qquad (31)$$

where α_l^* is the estimated vector of α at the lth experiment of the simulation.

The EMSE values of the OLS, KL, and GKL estimators are presented in **Tables 2**, **3**. We can conclude the following based on the simulation results:

1. When the standard deviation (σ), the degree of multicollinearity (ρ), and the explanatory variables number

(p) get an increase, the EMSE values of estimators get an increase.
2. The EMSE values of estimators get a decrease in case of the sample size gets an increase.
3. The GKL is better than the OLS estimator in all different values of factors except when $\sigma = 1$ and $\rho = 0.80, \ 0.90$ with the considered values of p and n.
4. The GKL is better than the KL estimator in all different values of factors except the following cases: (i) for $n = 50$ when $\sigma = 1$ and $\rho = 0.80,\ 0.90$ with $p = 3$ or 7, (ii) for $n = 100, 150$ when $\sigma = 1$ in all presented values of ρ with $p = 3$ or when $\sigma = 5$ and $\rho = 0.80$ with $p = 3$, and (iii) for $n = 100, 150$ when $\sigma = 1$ and $\rho = 0.80, \ 0.90$ with $p = 7$.
5. Finally, we see that the proposed GKL estimator is obviously efficient in case of the standard deviation getting large and when the correlation among the explanatory variables are very high.

EMPIRICAL APPLICATION

For clarifying the performance of the proposed GKL estimator, the dataset of the Portland cement that was originally due to Woods et al. [32], which was considered in Kibria and Lukman [1], where the dependent variable is the heat evolved after 180 days of curing and measured in calories per gram of cement. In this study, the first explanatory variable is tricalcium aluminate, the second explanatory variable is tricalcium silicate, the third explanatory variable is tetracalcium aluminoferrite, and the fourth explanatory variable is β-dicalcium silicate. The eigenvalues of $X'X$ matrix are 44,676.21, 5,965.42, 809.95, and 105.42. Then, the condition number is 20.58. Therefore, multicollinearity exists among the predictors. The estimated error variance is $\hat{\sigma}^2 = 5.84$, which shows high noise in the data. The estimated values of the optimal parameters in the GKL estimator are calculated as derived in Section Statistical Methodology. Also, the equation proposed by Kibria and Lukman [1] for estimating the biasing parameter of the KL estimator is used. Consequently, the mean square error (MSE) of the OLS, KL, and GKL estimators are presented in **Table 4**. From **Table 4**, we can note that the KL estimator is better than the OLS estimator, and the GKL estimator is better than the OLS and KL estimators.

CONCLUSION

In this paper, we proposed the GKL estimator. The performance of the proposed GKL estimator is theoretically compared with the OLS, GR, GL, and KL estimators in terms of known matrix mean squared error. Moreover, the optimal shrinkage parameter of the proposed GKL estimator is presented. A simulation study and the numerical example were performed for comparing the performance of the proposed GKL estimator with the OLS and KL estimators based on the estimated mean squared error criterion. The results indicate that the proposed estimator is better than other estimators, in particular, in the case the standard deviation of the errors was large and when the correlation between the explanatory variables is very high.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

ID, MA, and FA contributed to conception and structural design of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

ACKNOWLEDGMENTS

The authors would like to thank the Deanship of Scientific Research at King Saud University, represented by the Research Center, at CBA for supporting this research financially.

REFERENCES

1. Kibria BMG, Lukman AF. A new ridge-type estimator for the linear regression model: simulations and applications. *Hindawi Sci.* (2020) 2020:9758378. doi: 10.1155/2020/9758378
2. Hoerl AE, Kennard RW. Ridge regression: biased estimation for nonorthogonal problems. *Technometrics*. (1970) 12:55–67.
3. Liu K. A new class of biased estimate in linear regression. *Commun Stat Theory Methods.* (1993) 22:393–402.
4. Lukman AF, Algamal ZY, Kibria BG, Ayinde K. The KL estimator for the inverse Gaussian regression model. *Concurr Comput Prac Exp.* (2021) 33:e6222. doi: 10.1002/cpe.6222
5. Lukman AF, Dawoud I, Kibria BM, Algamal ZY, Aladeitan B. A new ridge-type estimator for the gamma regression model. *Scientifica*. (2021) 2021:5545356. doi: 10.1155/2021/5545356
6. Akram MN, Kibria BG, Abonazel MR, Afzal N. On the performance of some biased estimators in the gamma regression model: simulation and applications. *J Stat Comput Simul.* (2022) 1–23. doi: 10.1080/00949655.2022.2032059
7. Abonazel MR, Dawoud I, Awwad FA, Lukman AF. Dawoud–Kibria estimator for beta regression model: simulation and application. *Front Appl Math Stat.* (2022) 8:775068. doi: 10.3389/fams.2022.775068
8. Rashad NK, Hammood NM, Algamal ZY. Generalized ridge estimator in negative binomial regression model. *J Phys.* (2021) 1897:012019. doi: 10.1088/1742-6596/1897/1
9. Farghali RA, Qasim M, Kibria BM, Abonazel MR. Generalized two-parameter estimators in the multinomial logit regression model: methods, simulation and application. *Commun Stat Simul Comput.* (2021) 1–16. doi: 10.1080/03610918.2021.1934023
10. Abdulazeez QA, Algamal ZY. Generalized ridge estimator shrinkage estimation based on particle swarm optimization algorithm. *Electro J Appl Stat Anal.* (2021) 14:254–65. doi: 10.1285/I20705948V14N1P254
11. Wang SG, Wu MX, Jia ZZ. *Matrix Inequalities*. Beijing: Chinese Science Press (2006).
12. Farebrother RW. Further results on the mean square error of ridge regression. *J R Stat Soc Ser B.* (1976) 38:248–50.

13. Trenkler G, Toutenburg H. Mean squared error matrix comparisons between biased estimators-an overview of recent results. *Stat Pap.* (1990) 31: 165–79.
14. Hoerl AE, Kannard RW, Baldwin KF. Ridge regression: some simulations. *Commun. Stat.* (1975) 4:105–23.
15. Khalaf G, Shukur G. Choosing ridge parameter for regression problems. *Commun Stat Theory Methods.* (2005) 34:1177–82. doi: 10.1081/STA-200056836
16. Khalaf G, Månsson K, Shukur G. Modified ridge regression estimators. *Commun Stat Theory Methods.* (2013) 42:1476–87. doi: 10.1080/03610926.2011.593285
17. Månsson K, Kibria BMG, Shukur G. Performance of some weighted Liu estimators for logit regression model: an application to Swedish accident data. *Commun Stat Theory Methods.* (2015) 44:363–75. doi: 10.1080/03610926.2012.745562
18. Kibria BMG, Banik S. Some ridge regression estimators and their performances. *J Mod Appl Stat Methods.* (2016) 15:206–38. doi: 10.22237/jmasm/1462075860
19. Algamal ZY. A new method for choosing the biasing parameter in ridge estimator for generalized linear model. *Chemometr Intell Lab Syst.* (2018) 183:96–101. doi: 10.1016/j.chemolab.2018.10.014
20. Abonazel MR, Farghali RA. Liu-type multinomial logistic estimator. *Sankhya B.* (2019) 81:203–25. doi: 10.1007/s13571-018-0171-4
21. Qasim M, Amin M, Omer T. Performance of some new Liu parameters for the linear regression model. *Commun Stat Theory Methods.* (2020) 49:4178–96. doi: 10.1080/03610926.2019.1595654
22. Suhail M, Chand S, Kibria BG. Quantile based estimation of biasing parameters in ridge regression model. *Commun Stat Simul Comput.* (2020) 49:2732–44. doi: 10.1080/03610918.2018.1530782
23. Babar I, Ayed H, Chand S, Suhail M, Khan YA, Marzouki R. Modified Liu estimators in the linear regression model: an application to tobacco data. *PLoS ONE.* (2021) 16:e0259991. doi: 10.1371/journal.pone.0259991
24. Abonazel MR, Taha IM. Beta ridge regression estimators: simulation and application. *Commun Stat Simul Comput.* (2021) 1–13. doi: 10.1080/03610918.2021.1960373
25. McDonald GC, Galarneau DI. A Monte Carlo evaluation of some ridge-type estimators. *J Am Stat Assoc.* (1975) 70:407–16. doi: 10.2307/2285832
26. Gibbons DG. A simulation study of some ridge estimators. *J Am Stat Assoc.* (1981) 76:131–9.
27. Kibria BMG. Performance of some new ridge regression estimators. *Commun Stat Simul Comput.* (2003) 32:419–35. doi: 10.1081/SAC-120017499
28. Dawoud I, Abonazel MR. Robust Dawoud–Kibria estimator for handling multicollinearity and outliers in the linear regression model. *J Stat Comput Simul.* (2021) 91:3678–92. doi: 10.1080/00949655.2021.1945063
29. Algamal ZY, Abonazel MR. Developing a Liu-type estimator in beta regression model. *Concurr Comput Pract Exp.* (2022) 34:e6685. doi: 10.1002/cpe.6685
30. Abonazel MR, Algamal ZY, Awwad FA, Taha IM. A new two-parameter estimator for beta regression model: method, simulation, and application. *Front Appl Math Stat.* (2022) 7:780322. doi: 10.3389/fams.2021.780322
31. Awwad FA, Dawoud I, Abonazel MR. Development of robust Özkale–Kaçiranlar and Yang–Chang estimators for regression models in the presence of multicollinearity and outliers. *Concurr Comput Pract Exp.* (2022) 34:e6779. doi: 10.1002/cpe.6779
32. Woods H, Steinour HH, Starke HR. Effect of composition of Portland cement on heat evolved during hardening. *Indust Eng Chem.* (1932) 24:1207–14. doi: 10.1021/ie50275a002

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

ORIGINAL RESEARCH
published: 28 June 2022
doi: 10.3389/fams.2022.920186

On Two Localized Particle Filter Methods for Lorenz 1963 and 1996 Models

Nora Schenk [1,2]*, *Roland Potthast* [1,3] *and Anne Rojahn* [1,3]

[1] *Data Assimilation Unit, Deutscher Wetterdienst, Offenbach am Main, Germany,* [2] *Institute of Mathematics, Goethe-University Frankfurt, Frankfurt am Main, Germany,* [3] *Department of Mathematics, University of Reading, Reading, United Kingdom*

Nonlinear data assimilation methods like particle filters aim to improve the numerical weather prediction (NWP) in non-Gaussian setting. In this manuscript, two recent versions of particle filters, namely the Localized Adaptive Particle Filter (LAPF) and the Localized Mixture Coefficient Particle Filter (LMCPF) are studied in comparison with the Ensemble Kalman Filter when applied to the popular Lorenz 1963 and 1996 models. As these particle filters showed mixed results in the global NWP system at the German meteorological service (DWD), the goal of this work is to show that the LMCPF is able to outperform the LETKF within an experimental design reflecting a standard NWP setup and standard NWP scores. We focus on the root-mean-square-error (RMSE) of truth minus background, respectively, analysis ensemble mean to measure the filter performance. To simulate a standard NWP setup, the methods are studied in the realistic situation where the numerical model is different from the true model or the nature run, respectively. In this study, an improved version of the LMCPF with exact Gaussian mixture particle weights instead of approximate weights is derived and used for the comparison to the Localized Ensemble Transform Kalman Filter (LETKF). The advantages of the LMCPF with exact weights are discovered and the two versions are compared. As in complex NWP systems the individual steps of data assimilation methods are overlaid by a multitude of other processes, the ingredients of the LMCPF are illustrated in a single assimilation step with respect to the three-dimensional Lorenz 1963 model.

Keywords: data assimilation, particle filter, nonlinear systems, ensemble filter, Kalman filter, Lorenz 1963 system, Lorenz 1996 system

Edited by:
Lili Lei,
Nanjing University, China

Reviewed by:
Jie Feng,
Fudan University, China
Mengbin Zhu,
State Key Laboratory of Geo-information Engineering, China

***Correspondence:**
Nora Schenk
nora.schenk@dwd.de

Specialty section:
This article was submitted to Dynamical Systems, a section of the journal Frontiers in Applied Mathematics and Statistics

Received: *14 April 2022*
Accepted: *30 May 2022*
Published: *28 June 2022*

Citation:
Schenk N, Potthast R and Rojahn A (2022) On Two Localized Particle Filter Methods for Lorenz 1963 and 1996 Models. Front. Appl. Math. Stat. 8:920186. doi: 10.3389/fams.2022.920186

1. INTRODUCTION

Data assimilation methods combine numerical models and observations to generate an improved state estimate. Besides optimization approaches, ensemble methods use an ensemble of states to approximate underlying probability distributions. For example the ensemble Kalman filter presented in Evensen [1] (see also [2, 3]) carries out Bayesian state estimation and samples from Gaussian distributions which equals a linearity assumption of the underlying system. However, the local ensemble transform Kalman filter (LETKF; [4]) is widely used in high dimensional environments. For example, the LETKF is successfully used as ensemble data assimilation method in the numerical weather prediction (NWP) system at the German meteorological service (DWD). Nevertheless, there is the aim to develop more general ensemble methods to account for the increasing complexity of numerical models.

Particle filter methods are based on Monte Carlo schemes and aim to solve the nonlinear filtering problem without any further assumptions on the distributions. Since Monte Carlo methods suffer the curse of dimensionality, the application of classical or bootstrap particle filters to high- dimensional problems results in filter divergence or filter collapse (see [5–7]). After first attempts to carry out nonlinear Bayesian state estimation approximately by Gordon et al. [8], further particle filters are developed, which are able to overcome filter collapse. For an overview of particle filters we refer to van Leeuwen [5] and van Leeuwen et al. [9].

One idea to prevent filter collapse is to develop hybrid methods between particle filters and ensemble Kalman filters. Examples for hybrid filters are the adaptive Gaussian mixture filters [10], the ensemble Kalman particle filter [11], which is further developed in Robert and Künsch [12] and Robert et al. [13], the merging particle filter [14] and the nonlinear ensemble transform filter (e.g., [15, 16]) which resembles the ensemble transform Kalman filter [17]. Transportation particle filters follow the approach to use transformations to transport particles in a deterministic way. A one-step transportation is carried out in Reich [18] and tempering of the likelihood, which leads to a multi-step transportation, is presented in, e.g., Neal [19], Del Moral et al. [20], Emerick and Reynolds [21], and Beskos et al. [22]. The guided particle filter described in van Leeuwen et al. [23] and van Leeuwen [5] tempers in the time domain, which means that background particles at each time step between two observations are used. The transportation of particle filters can also be described by differential equations. In Reich [24] and Reich and Cotter [25], the differential equation is simulated using more and more tempering steps. Approximations to the differential equation can also be derived by Markov-Chain Monte Carlo methods [25–27]. Localization is another approach in particle filter methods to overcome filter collapse. Localization schemes based on resampling are used in e.g., the local particle filter [28] which is applied for mesoscale weather prediction [29]. Additionally, the local particle filter (LPF) [30], the localized adaptive particle filter (LAPF; [31]) and the localized mixture coefficients particle filter (LMCPF; [32]) are based on localization schemes.

Moreover, the localized mixture coefficients particle filter (LMCPF) is based on Gaussian mixture distributions. In 1972, Alspach and Sorenson already introduced an approach to nonlinear Bayesian estimation using Gaussian sum approximations combined with linearization ideas [33]. Anderson and Anderson first presented a Monte Carlo approach with prior approximation by Gaussian or sum of Gaussian kernels in geophysical literature [34]. They proposed to extend the presented kernel filter by the transformation of the equations to a subspace spanned by the ensemble members to apply the filter in high-dimensional systems. The LMCPF is based on this kind of transformation. The first attempts were followed from various approaches to filtering with the usage of Gaussian mixture distributions (e.g., [35–38]). Some of the particle filters mentioned above are based on Gaussian mixture distributions as well (e.g., [10, 11, 24]).

The localized particle filter methods LPF [30], LAPF and LMCPF are structured in a way that a consistent implementation in existing LETKF code is possible. In Kotsuki et al. [39], the LPF and its Gaussian mixture extension, which resembles the LMCPF, are tested in an intermediate AGCM (SPEEDY model). Moreover, LAPF and LMCPF are applied in the global NWP system at DWD (see [31, 32]). The comparison of the LMCPF to the LETKF for the global ICON model [40] yields mixed results. In this study, we investigate if the LMCPF can outperform the LETKF with respect to a standard NWP setup and standard NWP score in the dynamical systems Lorenz 1963 and Lorenz 1996. We will see later that the answer is indeed positive and that the LMCPF yields far better results than the LAPF. To this end, a model error is introduced by applying different model parameters for the true run and in the forecast step. Furthermore, we focus on the root-mean-square-error of truth minus background, respectively, analysis ensemble mean, which is an important score in NWP, rather than looking at an entire collection of measures. In this study, we present and apply a revised version of the LMCPF. We derive the exact Gaussian particle weights, which are then used in the resampling step instead of approximate weights. This promising completion of the method was also recently introduced in Kotsuki et al. [39] and tested for an intermediate AGCM model. We will see that the revised method leads to the survival of a larger selection of background particles and as a consequence thereof to a higher filter stability concerning the spread control parameters.

In addition, the individual ingredients of the LMCPF method are portrayed in one assimilation step with respect to the Lorenz 1963 model. Background and analysis ensemble as well as the true state and observation vector can be easily displayed for this three dimensional model. With this part, we want to illustrate the advantage of LMCPF compared to LAPF in the case that the observation is far away from the ensemble. Furthermore, the difference between the approximate and exact particle weights are discussed and the improvement of LMCPF over LETKF for a bimodal background distribution is shown.

The manuscript is structured as follows. Section 2 covers the experimental setup based on the dynamical systems Lorenz 1963 and Lorenz 1996. The three localized ensemble data assimilation methods LMCPF, LAPF and LETKF are mathematically described in Section 3, which includes the derivation of the exact particle weights for the LMCPF. In Section 4, the LMCPF is studied for one assimilation step with respect to the Lorenz 1963 model. Finally, LMCPF is compared to LETKF and LAPF for Lorenz 1963 and Lorenz 1996 in Section 5 and the conclusion follows in Section 6.

2. EXPERIMENTAL SETUP: LORENZ MODELS

The mathematician Edward Lorenz first presented the chaotic dynamical systems Lorenz 1963 and 1996. These are frequently used to develop and test data assimilation methods in a well understood and controllable environment. This section aims to state the experimental setup.

2.1. Lorenz 1963 Model

In Lorenz [41], Edward Lorenz introduced a nonlinear dynamical model, which is denoted as Lorenz 1963. Due to its chaotic behavior, the system has become a popular toy model to investigate and compare data assimilation methods (e.g., [34, 38, 42]).

The dynamics of Lorenz 1963 represent a simplified version of thermal convection. The three coupled nonlinear differential equations are given by

$$\frac{dx_1}{dt} = \sigma(x_2 - x_1) \tag{1}$$

$$\frac{dx_2}{dt} = \rho x_1 - x_2 - x_1 x_3 \tag{2}$$

$$\frac{dx_3}{dt} = x_1 x_2 - \beta x_3 \tag{3}$$

where $x_1(t)$, $x_2(t)$, and $x_3(t)$ are the prognostic variables and σ, ρ, and β denote the parameters of the model. In terms of the physical interpretation, σ is the Prandtl number, ρ a normalized Rayleigh number and β a non-dimensional wave number (see [43]). In this work, we follow Lorenz' suggestion to set $\sigma = 10$, $\rho = 28$ and $\beta = 8/3$, for which the system shows chaotic behavior [41]. In case of this parameter setting, the popular butterfly attractor is obtained (see **Figure 5**). Furthermore, x_1 describes the intensity of the convective motion, x_2 the temperature difference between the ascending and descending currents and the last variable x_3 denotes the distortion of the vertical temperature profile from linearity [41].

2.2. Lorenz 1996 Model

Since the introduction of the Lorenz 1996 model in Lorenz [44], the dynamical system is used as popular test bed for data assimilation methods (e.g., [28, 36, 45]). Not only different adaptions of the ensemble Kalman filter but also particle filter schemes or hybrid methods combining particle filter and EnKF schemes are tested in the high-dimensional and chaotic environment given by Lorenz 1996 with specific parameter settings (e.g., [30, 46, 47]). In contrast to Lorenz 1963, localization is an important component of the investigation of data assimilation methods and the later Lorenz 1996 model invites to test localization schemes (e.g., [48]).

The model considers $n \in \mathbb{N}$ coupled time-dependent variables, whose dynamics are described by a system of n ordinary differential equations. We consider the state variable as $x(t) = (x^{(1)}(t), \dots, x^{(n)}(t)) \in \mathbb{R}^n$ for $t \in \mathbb{R}_+$. The dynamics of the N-th component are governed by the ordinary differential equation

$$\frac{\mathrm{d}x^{(N)}}{\mathrm{d}t} = -x^{(N-2)}x^{(N-1)} + x^{(N-1)}x^{(N+1)} - x^{(N)} + F \tag{4}$$

where the constant F is independent of N and describes a forcing term. Furthermore, we define

$$x^{(N-n)} := x^{(N)} \tag{5}$$

$$x^{(N+n)} := x^{(N)} \tag{6}$$

so that Equation (4) is valid for any $N = 1, \dots, n$. In addition to the external forcing term, the linear terms describe internal dissipation whereas the nonlinear, respectively, quadratic terms simulate advection. In this study, we use $F = 8$ as forcing term for the true run and choose differing values for the model propagation step.

In a meteorological context, each variable represents an atmospheric quantity, e.g., temperature, at one longitude on a latitude circle. The equidistant distribution of the nodes on a latitude circle for $n = 40$ variables is illustrated in **Figure 1**.

2.3. Data Assimilation Setup

To test data assimilation methods with the Lorenz models, observations are produced at equidistant distributed measurement times. The system of differential equations of Lorenz 1963 model, respectively, Lorenz 1996 model is solved by a fourth-order Runge-Kutta scheme using a time-step of 0.05. The integration over a certain time is stored as truth, from which observations are generated with a distance of Δ_t time units. The true run is performed with model parameters $\sigma^{\text{true}} = 10$, $\rho = 28$ and $\beta = 8/3$ for Lorenz 1963 and with the forcing term $F^{\text{true}} = 8$ for the 40-dimensional Lorenz 1996 model. The integration of the ensemble of states is accomplished for different model parameters σ for Lorenz 1963 and F for Lorenz 1996 in order to simulate model error. Furthermore, the observation operator $H \in \mathbb{R}^{m \times n}$ is chosen linear for both dynamical systems. The observation vector y_k at the $k-$th measurement at time t_k is defined by

$$y_k = H \cdot x_k^{\text{true}} + \eta \in \mathbb{R}^m \tag{7}$$

whereas the entries of $\eta \in \mathbb{R}^m$ are randomly drawn from a Gaussian distribution with zero expectation and standard deviation σ_{obs}. Additionally, the observation error covariance matrix is represented by

$$R = \sigma_{\text{obs}}^2 \cdot I_m \in \mathbb{R}^{m \times m} \tag{8}$$

with the $m \times m$-identity matrix I_m. The ensemble is initialized by random draws from a uniform distribution around the true starting point x_0^{true}.

3. LOCALIZED ENSEMBLE DATA ASSIMILATION METHODS

Data assimilation methods aim to estimate some state vector. Methods based on an ensemble of states can additionally estimate the uncertainty of the state and provide an idea for the associated distribution. This section covers three localized ensemble data assimilation methods, which are compared against each other later in this paper. The localized adaptive particle filter (LAPF; [31]) describes a particle filter method which is applicable to real-size numerical weather prediction and implemented in the system of the German meteorological service (DWD). To improve the method and approximate the scores, the LAPF was further developed, which resulted in the localized mixture coefficients particle filter (LMCPF). The LMCPF combines a

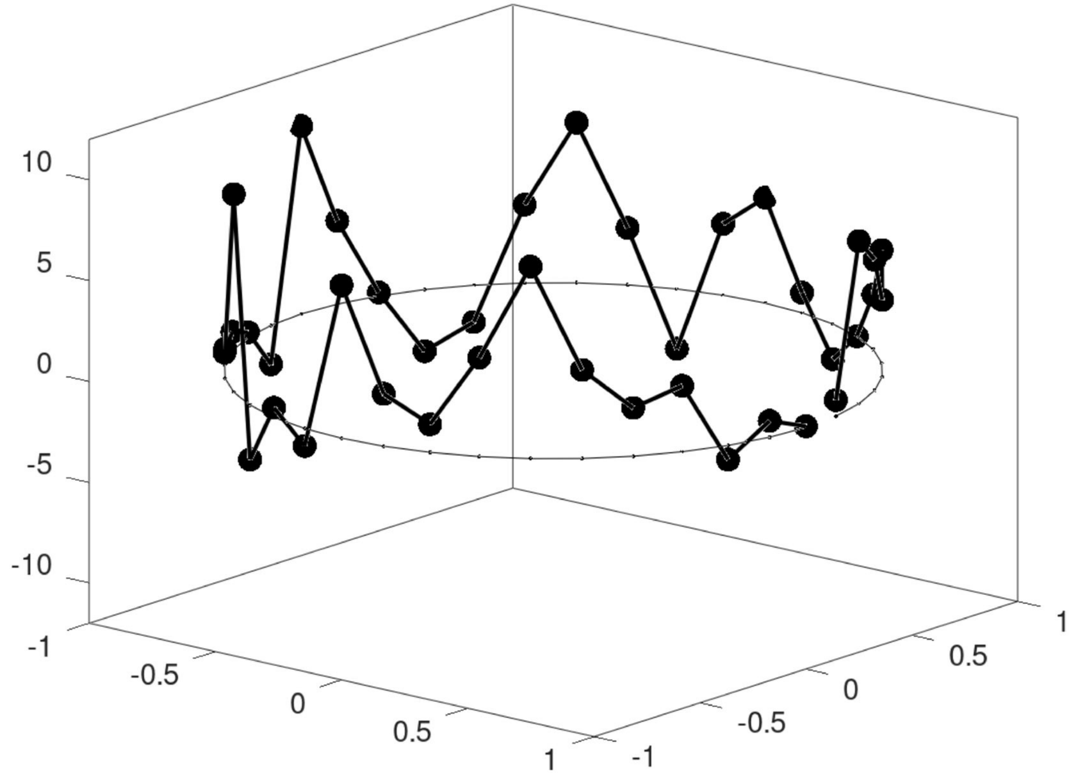


FIGURE 1 | Set-up for Lorenz 1996 model with $n = 40$ variables.

resampling step following the Monte Carlo approach with a shift of the particles toward the observation. The shift results from the application of Gaussian (mixture) distributions and exists in the localized ensemble transform Kalman filter (LETKF) [4] in the form that the ensemble mean is shifted. The LETKF is widely used in the data assimilation community and therefore already improved. Due to similarities between LETKF and the particle filter methods LAPF and LMCPF, the ensemble Kalman filter represents a good method to compare the newer methods LAPF and LMCPF with.

All of these ensemble methods fulfill Bayes' theorem in approximation. With the aid of Bayes' formula, a given prior or background distribution can be combined with the so-called likelihood distribution to obtain a posterior or analysis distribution. In terms of probability density functions, the theorem yields

$$p^{(a)}(x) = c_a p(y|x) p^{(b)}(x), \qquad x \in \mathbb{R}^n, y \in \mathbb{R}^m \tag{9}$$

for the prior probability density function (pdf) $p^{(b)} : \mathbb{R}^n \to [0, \infty)$, the likelihood pdf $p(\cdot|x) : \mathbb{R}^m \to [0, \infty)$ for $x \in \mathbb{R}^n$ and the resulting posterior pdf $p^{(a)} : \mathbb{R}^n \to [0, \infty)$ with $n, m \in \mathbb{N}$. In realistic NWP, the model space dimension $n \in \mathbb{N}$ is in general larger than the dimension of the observation space described by $m \in \mathbb{N}$. Furthermore, the constant $c_a \in \mathbb{R}$ in Equation (9) ensures that the resulting function is again a pdf. Due to the normalization constant, the likelihood function does not necessarily have to be a pdf to satisfy Bayes' formula. This form of Bayes' theorem is derived from the formula of the density function of a conditional probability function which is proven in Section 4-4 of Papoulis and Pillai [49].

In data assimilation, the likelihood is given by the observation error pdf as function of $x \in \mathbb{R}^n$ for given observation vector $y \in \mathbb{R}^m$. We assume a Gaussian distributed observation error for all presented filters, i.e.,

$$p(y|x) = \frac{1}{\sqrt{(2\pi)^m \det(R)}} \cdot \exp\left(-\frac{1}{2}(y - Hx)^T R^{-1} (y - Hx)\right), \tag{10}$$

for $x \in \mathbb{R}^n$, some observation vector $y \in \mathbb{R}^m$, the linear observation operator $H : \mathbb{R}^n \to \mathbb{R}^m$ and the observation error covariance matrix $R \in \mathbb{R}^{m \times m}$. The derivations of the following methods are carried out for a time-constant linear observation operator H. The assumption on the prior distribution differs for the filters. In the LAPF, the prior pdf is approximated by a sum of delta functions following the idea of the classical particle filter. The LMCPF assumes a sum of Gaussian kernels while the LETKF approximates the prior pdf by a Gaussian pdf.

All of the following methods are based on localization so that the steps are carried out locally at a series of analysis points. Furthermore, the observations are weighted depending on the distance to the current location. As Lorenz 1963 is only built on three variables, localization is not implemented for this model.

For the Lorenz 1996 model, the implementation is based on the smallest distance between two variables along the circle (e.g., [50]), which is plotted in **Figure 1**. The distances are weighted by the fifth-order polynomial localization Gaspari-Cohn function described in Gaspari and Cohn [51]. Moreover, the function depends from the localization radius $r_{\text{loc}} > 0$. The resulting weight matrix is applied by the Schur-product to the observation error covariance matrix R, which is then used to derive the analysis ensemble by one of the following methods.

In addition, the equations of the following localized methods are carried out in ensemble space to reduce the dimension. The ensemble space is spanned by the columns of

$$X := \left(x^{(b,1)} - \bar{x}^{(b)}, x^{(b,2)} - \bar{x}^{(b)}, \dots, x^{(b,L)} - \bar{x}^{(b)}\right) \in \mathbb{R}^{n\times L} \quad (11)$$

with ensemble size $L \in \mathbb{N}_{>1}$, respectively

$$Y := \left(y^{(b,1)} - \bar{y}^{(b)}, y^{(b,2)} - \bar{y}^{(b)}, \dots, y^{(b,L)} - \bar{y}^{(b)}\right) \in \mathbb{R}^{m\times L} \quad (12)$$

where $\bar{x}^{(b)}$ and $\bar{y}^{(b)}$ denote the mean of the background ensemble $(x^{(b,l)})_{l=1,\dots,L}$, i.e.,

$$\bar{x}^{(b)} = \frac{1}{L}\sum_{l=1}^{L} x^{(b,l)} \quad (13)$$

respectively the mean of the ensemble in observation space

$$\bar{y}^{(b)} = \frac{1}{L}\sum_{l=1}^{L} y^{(b,l)}. \quad (14)$$

The ensemble in observation space is obtained by the application of the observation operator H to the background ensemble, i.e.,

$$y^{(b,l)} := Hx^{(b,l)}, \qquad l = 1, \dots, L. \quad (15)$$

The orthogonal projection P onto the ensemble space span(Y) weighted by R^{-1} is defined as

$$P := Y(Y^*Y)^{-1}Y^* = Y(Y^TR^{-1}Y)^{-1}Y^TR^{-1} \quad (16)$$

whereas

$$Y^* = Y^TR^{-1} \quad (17)$$

denotes the adjoint of Y with respect to the weighted scalar product $< \cdot, \cdot >_{R^{-1}}$ on $\mathbb{R}^m$ and the standard scalar product on $\mathbb{R}^L$. To ensure the invertibility of Y^*Y, the formulas are restricted to $\mathcal{C}(Y^*)$ – the column space or range of Y^* – which is a subset of $N(Y)^\perp \subset \mathbb{R}^L$ (see Lemma 3.2.1 and Lemma 3.2.3 in Nakamura and Potthast [52]). Additionally, the matrix Y^*Y is denoted as

$$A := Y^*Y = Y^TR^{-1}Y. \quad (18)$$

3.1. Localized Adaptive Particle Filter

The LAPF, introduced in Potthast et al. [31], is based on the idea for classical particle filters (e.g., the Sequential Importance Resampling Filter by Gordon et al. [8]) to approximate the background distribution by a sum of delta distributions. Let $x^{(b,l)}$ for $l = 1, \dots, L$ be an ensemble of background particles with ensemble size $L \in \mathbb{N}_{>1}$. The background pdf is described by

$$p^{(b)}(x) := \frac{1}{L}\sum_{l=1}^{L} \delta(x - x^{(b,l)}). \quad (19)$$

With Bayes' Theorem for pdfs in Equation (9) and the observation error pdf $p(y|x)$, the posterior pdf results in

$$p^{(a)}(x) = c^{(a)}\sum_{l=1}^{L} p(y|x)\delta(x - x^{(b,l)}) \quad (20)$$

with the normalization factor $c^{(a)} \in \mathbb{R}_{>0}$. Following Anderson and Anderson [34], the relative probability p_l that a sample should be taken from the l-th summand of $p^{(a)}$ in the resampling step, is derived by

$$p_l = \frac{\int c^{(a)}p(y|x)\delta(x - x^{(b,l)})\,\mathrm{d}x}{\int p^{(a)}(x)\,\mathrm{d}x} = \frac{p(y|x^{(b,l)})}{\sum_{l=1}^{L} p(y|x^{(b,l)})}, \quad (21)$$

for $l = 1, \dots, L$. With the choice of a normal distributed observation error (Equation 10) this leads to

$$p_l = \frac{e^{-\frac{1}{2}(y-Hx^{(b,l)})^TR^{-1}(y-Hx^{(b,l)})}}{\sum_{l=1}^{L} e^{-\frac{1}{2}(y-Hx^{(b,l)})^TR^{-1}(y-Hx^{(b,l)})}} \quad (22)$$

as the normalization factor in Equation (10) does not depend on l and can be canceled. To resample from the posterior distribution, stratified resampling is performed in ensemble space. To this end, the weights

$$\tilde{w}^{(l)} := e^{\frac{1}{2}(y-Hx^{(b,l)})^TR^{-1}(y-Hx^{(b,l)})}, \qquad l = 1, \dots, L \quad (23)$$

are transformed to ensemble space with the help of the orthogonal projection P defined in Equation (16). With an analogous approach as in Section 3.2.1, the weights in ensemble space yield

$$\tilde{w}_{\text{ens}}^{(l)} = e^{-\frac{1}{2}(C-e_l)^TA(C-e_l)} \quad (24)$$

for $l = 1, \dots, L$ with $A = Y^TR^{-1}Y$ and the projected observation vector

$$C = A^{-1}Y^TR^{-1}(y - \bar{y}^{(b)}). \quad (25)$$

A detailed derivation of the weights in ensemble space is given in Potthast et al. [31]. These weights are normalized to obtain the relative weights

$$\tilde{w}^{(a,l)} = L \cdot \frac{\tilde{w}_{\text{ens}}^{(l)}}{\sum_{l=1}^{L} \tilde{w}_{\text{ens}}^{(l)}}, \qquad l = 1, \dots, L \quad (26)$$

which sum up to L. As next step, stratified resampling [53] is performed based on the ensemble weights. To this end, accumulated weights are calculated. For $l = 1, \dots, L$, the accumulated weights are defined by

$$w_{ac_0} = 0, \quad w_{ac_l} = w_{ac_{l-1}} + \tilde{w}^{(a,l)}. \tag{27}$$

Additionally, L on the interval $[0, 1]$ uniformly distributed random numbers r_l are generated to introduce the variable $R_l = l{-}1{+}r_l$ for $l = 1, \dots, L$. Then, the stratified resampling approach yields a matrix $\breve{W} \in \{0, 1\}^{L\times L}$ with entries

$$\breve{W} = \begin{cases} 1, & R_l \in (w_{ac_{i-1}}, w_{ac_i}] \\ 0, & \text{else} \end{cases} \tag{28}$$

where the number of ones in the i-th row indicates how often the i-th particle is chosen.

The particles chosen in the stratified resampling step build an ensemble of the background particles, which can be contained multiple times. To increase the ensemble variation, new particles are drawn from a Gaussian mixture distribution. Let each chosen particle represent the expectation of a Gaussian distribution with covariance $\sigma(\rho)^2/(L-1)\cdot I_L \in \mathbb{R}^{L\times L}$. Under allowance of the frequency, new particles are drawn from the Gaussian distribution. The covariance matrix equals the estimated background covariance matrix in ensemble space $B_{\text{ens}} = 1/(L-1)\cdot I_L \in \mathbb{R}^{L\times L}$ multiplied with an inflation factor $\sigma(\rho)$. The inflation factor is a rescaled version of the adaptive inflation factor ρ which is used in the LETKF (see [4]). The parameter ρ is defined by Equations (86) and (87). The dependence of $\sigma(\rho)$ on ρ is given by Equation (88). The detailed description is given in Potthast et al. [31] and in Section 3.2.3.

All in all, the steps can be combined in a matrix W_{LAPF}. Let $Z \in \mathbb{R}^{L\times L}$ be a matrix whose entries originate from a standard normal distribution. Together with the resampling matrix $\breve{W}$, the matrix W_{LAPF} is defined by

$$W_{\text{LAPF}} = \breve{W} + \frac{\sigma(\rho)}{\sqrt{L-1}} \cdot Z. \tag{29}$$

The full analysis ensemble is calculated by multiplication of the background ensemble with the matrix W_{LAPF}, i.e.,

$$(x^{(a,l)})_{l=1,\dots,L} = \bar{x}^{(b)} \cdot \mathbb{1} + X \cdot W_{\text{LAPF}} \tag{30}$$

where X describes the ensemble pertubation matrix defined in Equation (11) and $\mathbb{1} \in \mathbb{R}^{1\times L}$ denotes a row vector with ones as entries. The multiplication of background mean with $\mathbb{1}$ results in a matrix of size $n \times L$ with the mean vector replicated in each of the L columns.

3.2. Localized Mixture Coefficients Particle Filter

The LMCPF, presented in Walter et al. [32], builds on the LAPF but differs in the assumption on the background distribution. In difference to the ansatz of classical particle filters, the background particles are interpreted as the mean of Gaussian distributions. The background pdf is described as the sum of these Gaussians where each distribution has the same covariance matrix, i.e.,

$$p^{(b)}(x) := c^{(b)} \sum_{l=1}^{L} e^{-\frac{1}{2}(x-x^{(b,l)})^T B^{-1}(x-x^{(b,l)})} \tag{31}$$

with ensemble size $L \in \mathbb{N}_{>1}$ and the normalization factor

$$c^{(b)} := \frac{1}{L \cdot \sqrt{(2\pi)^n \det(B)}}. \tag{32}$$

The covariance matrix is estimated by the background particles, i.e.,

$$B := \gamma X X^T \tag{33}$$

with the ensemble pertubation matrix X defined in Equation (11) and the parameter

$$\gamma = \frac{\kappa}{L-1} \in \mathbb{R}_+. \tag{34}$$

With the parameter κ, the background uncertainty can be controlled. The general covariance estimator is given for $\kappa = 1$. To ensure the invertibility of B, the formulas are restricted to $\mathcal{C}(X)$ – the range of X. From definition (Equation 33) the covariance matrix in ensemble space is derived by

$$B_{\text{ens}} = \gamma I_L \in \mathbb{R}^{L\times L} \tag{35}$$

with the identity matrix $I_L \in \mathbb{R}^{L\times L}$. Following Bayes' Theorem, the analysis pdf is given by

$$p^{(a)}(x) := p(y|x) \cdot p^{(b)}(x) = \tilde{c}^{(a)} \sum_{l=1}^{L} p(y|x) \cdot p^{(b,l)}(x) \tag{36}$$

where $p^{(b,l)}(x)$ denotes the l-th summand of the background pdf in Equation (31). The likelihood $p(y|x)$ is chosen as Gaussian (see Equation 10). Following Theorem 4.1 in Anderson and Moore [54], the analysis pdf can be explicitly calculated. The result is again a Gaussian mixture pdf, i.e.,

$$p^{(a)}(x) = c^{(a)} \sum_{l=1}^{L} w^{(l)} \cdot e^{\left(-\frac{1}{2}(x-x^{(a,l)})^T (B^{(a)})^{-1}(x-x^{(a,l)})\right)} \tag{37}$$

with

$$B^{(a)} := (B^{-1} + H^T R^{-1} H)^{-1} \tag{38}$$

$$x^{(a,l)} := x^{(b,l)} + B^{(a)} H^T R^{-1}(y - Hx^{(b,l)}) \tag{39}$$

$$w^{(l)} := e^{\left(-\frac{1}{2}(y-Hx^{(b,l)})^T \gamma^{-1}(\gamma^{-1}R+YY^T)^{-1}(y-Hx^{(b,l)})\right)} \tag{40}$$

and a normalization factor $c^{(a)}$ such that the integral of $p^{(a)}(x)$ over the range of X denoted by $\mathcal{C}(X)$ yields one. The weights $w^{(l)}$ are important to obtain a sample from the posterior distribution. The relative probability that a sample from the l-th summand of

$p^{(a)}$ should be taken is described in Anderson and Anderson [34] by

$$p_l = \frac{\int c^{(a)} w^{(l)} \cdot e^{\left(-\frac{1}{2}(x-x^{(a,l)})^T (B^{(a)})^{-1} (x-x^{(a,l)})\right)} \, \mathrm{d}x}{\int p^{(a)}(x) \, \mathrm{d}x} = \frac{w^{(l)}}{\sum_{l=1}^{L} w^{(l)}}. \quad (41)$$

With the following steps, a posterior ensemble is generated as a sample of the posterior distribution in Equation (37).

3.2.1. Stratified Resampling

In the original version of the LMCPF described in Walter et al. [32], the particle weights are approximated by those of the LAPF defined in Equation (23). In this work, the exact Gaussian mixture weights are derived and applied in the resampling step. Furthermore, the effect on the filter performance is discovered. In Kotsuki et al. [39], the exact weights are applied to the Gaussian mixture extension of the LPF [30] and an improvement of the stability of the method is detected with respect to the inflation parameters within an intermediate AGCM.

To reduce the dimensionality, the weights in Equation (40) are transformed and projected in ensemble space. To this end, the sum of the projection P defined in Equation (16) and $I - P$ with the identity matrix I is applied to the exponent of Equation (40). The weights are transformed to

$$\begin{aligned} w^{(l)} &= e^{\left(-\frac{1}{2}([P+(I-P)](y-Hx^{(b,l)}))^T \gamma^{-1} (\gamma^{-1}R+YY^T)^{-1} [P+(I-P)](y-Hx^{(b,l)})\right)} \quad (42) \\ &= c_{I-P} \cdot e^{\left(-\frac{1}{2}(y-Hx^{(b,l)})^T P^T \gamma^{-1} (\gamma^{-1}R+YY^T)^{-1} P(y-Hx^{(b,l)})\right)} \quad (43) \end{aligned}$$

whereas c_{I-P} is defined by

$$c_{I-P} := e^{\left(-\frac{1}{2}(y-Hx^{(b,l)})^T (I-P)^T \gamma^{-1} (\gamma^{-1}R+YY^T)^{-1} (I-P)(y-Hx^{(b,l)})\right)}. \quad (44)$$

First, the observation minus first guess vector can be reshaped to

$$y - Hx^{(b,l)} = (y - \bar{y}^{(b)}) + (\bar{y}^{(b)} - Hx^{(b,l)}) = y - \bar{y}^{(b)} - Ye_l \quad (45)$$

with the l-th unit vector $e_l \in \mathbb{R}^L$. The application of the projection matrix to Equation (45) leads to

$$P(y - Hx^{(b,l)}) = YA^{-1}Y^T R^{-1}[(y - \bar{y}^{(b)}) - Ye_l] = Y(C - e_l) \quad (46)$$

whereas C denotes the projected observation vector in ensemble space

$$C := A^{-1}Y^T R^{-1}(y - \bar{y}^{(b)}). \quad (47)$$

With the aid of Equation (45), the application of $I - P$ to observation minus first guess vector yields

$$\begin{aligned} (I-P)(y - Hx^{(b,l)}) &= (I-P)(y - \bar{y}^{(b)}) - (I-P)Ye_l \quad (48) \\ &= (y - \bar{y}^{(b)}) - YA^{-1}Y^T R^{-1}(y - \bar{y}^{(b)}) \\ &\quad - Ye_l + YA^{-1}Y^T R^{-1} Ye_l \quad (49) \\ &= (y - \bar{y}^{(b)}) - YC. \quad (50) \end{aligned}$$

This expression do not depend on l so that c_{I-P} of Equation (44) is constant and has no impact on the relative weights of the particles [see Equation (43)]. To derive the transformation, the equality

$$Y^T(\gamma^{-1}R + YY^T)^{-1} = (\gamma^{-1}I + Y^T R^{-1} Y)^{-1} Y^T R^{-1} \quad (51)$$

is used. Equation (51) is shown by multiplying

$$(\gamma^{-1}I + Y^T R^{-1} Y)Y^T = Y^T R^{-1}(\gamma^{-1}R + YY^T) \quad (52)$$

from the left with the inverse

$$(\gamma^{-1}I + Y^T R^{-1} Y)^{-1} \quad (53)$$

and from the right with the inverse matrix

$$(\gamma^{-1}R + YY^T)^{-1} = R^{-1}(\gamma^{-1}I + YY^T R^{-1})^{-1}. \quad (54)$$

The invertibility of $\gamma^{-1}I + Y^*Y$ and $\gamma^{-1}I + YY^*$ on $N(Y)^\perp$, respectively, $\mathcal{C}(Y)$ follows from Theorem 3.1.8 in Nakamura and Potthast [52]. Y^* denotes the adjoint matrix defined in Equation (17). The first mixed term

$$\begin{aligned} &(P(y - Hx^{(b,l)}))^T \gamma^{-1}(\gamma^{-1}R + YY^T)^{-1}(I - P)(y - Hx^{(b,l)}) \quad (55) \\ &= (y - Hx^{(b,l)})^T P^T \gamma^{-1}(\gamma^{-1}R + YY^T)^{-1}(y - Hx^{(b,l)}) \quad (56) \\ &\quad -(y - Hx^{(b,l)})^T P^T \gamma^{-1}(\gamma^{-1}R + YY^T)^{-1} P(y - Hx^{(b,l)}) \quad (57) \end{aligned}$$

reduce to zero if the equality

$$P^T(\gamma^{-1}R + YY^T)^{-1} = P^T(\gamma^{-1}R + YY^T)^{-1}P \quad (58)$$

holds. Starting with the right hand side of the equation, we obtain

$$\begin{aligned} P^T(\gamma^{-1}R + YY^T)^{-1}P &= R^{-1}YA^{-1}(\gamma^{-1}I \\ &\quad + Y^T R^{-1} Y)^{-1} Y^T R^{-1} Y A^{-1} Y^T R^{-1} \quad (59) \\ &= R^{-1}YA^{-1}(\gamma^{-1}I + Y^T R^{-1} Y)^{-1} Y^T R^{-1} \quad (60) \\ &= P^T(\gamma^{-1}R + YY^T)^{-1} \quad (61) \end{aligned}$$

with the application of equality [Equation (51)] in the first and last step and the definition of A in Equation (18) in the second step. The reduction of the second mixed term to zero can be proven following an analog approach. The combination of the formulation in Equation (46) with Equation (51) leads to the exponent

$$\begin{aligned} &(P(y - Hx^{(b,l)}))^T \gamma^{-1}(\gamma^{-1}R + YY^T)^{-1} P(y - Hx^{(b,l)}) \quad (62) \\ &= (C - e_l)^T Y^T \gamma^{-1}(\gamma^{-1}R + YY^T)^{-1} Y(C - e_l) \quad (63) \\ &= (C - e_l)^T \gamma^{-1}(\gamma^{-1}I + Y^T R^{-1} Y)^{-1} Y^T R^{-1} Y(C - e_l) \quad (64) \end{aligned}$$

Finally, the particle weights in ensemble space yield

$$w_{\text{ens}}^{(l)} = e^{-\frac{1}{2}(C-e_l)^T \gamma^{-1}(\gamma^{-1}I+A)^{-1}A(C-e_l)}, \qquad l = 1, \ldots, L. \quad (65)$$

with the relation $w_{\text{ens}}^{(l)} = c_{I-P} \cdot w^{(l)}$ to the weights in model space with c_{I-P} defined in Equation (44). In the following, the normalized weights

$$w^{(a,l)} = L \cdot \frac{w_{\text{ens}}^{(l)}}{\sum_{l=1}^{L} w_{\text{ens}}^{(l)}}, \qquad l = 1, \ldots, L \tag{66}$$

are used which sum up to L.

Following the approach of stratified resampling [53], uniformly distributed random numbers are used to calculate the frequency of each particle with the aid of the respective accumulated weights. For $l = 1, \ldots, L$, the accumulated weights are defined by

$$w_{ac_0} = 0, \quad w_{ac_l} = w_{ac_{l-1}} + w^{(a,l)}. \tag{67}$$

Then, L on the interval $[0, 1]$ uniformly distributed random numbers r_l are generated to introduce the variable $R_l = l - 1 + r_l$ for $l = 1, \ldots, L$. The approach of stratified resampling then leads to the matrix $\breve{W} \in \{0, 1\}^{L \times L}$ with entries

$$\breve{W} = \begin{cases} 1, & R_l \in (w_{ac_{i-1}}, w_{ac_i}] \\ 0, & \text{else} \end{cases} \tag{68}$$

where the number of ones in the i-th row indicates how often the i-th particle is chosen.

3.2.2. Shift of Particles

Compared to the LAPF, the Gaussian mixture representation leads to a shift of the particles toward the observation. The shift resembles the shift of the mean of all particles toward the observation in ensemble space in the LETKF (see [4]). The new location of the particles is described by the expectation vectors in Equation (39) of the kernels of the posterior Gaussian mixture distribution. To carry out the particle shift, the transformed formula of Equation (39) is derived. First, the representation of the analysis covariance matrix $B^{(a)}$ defined in Equation (38) is derived. To this end, the analysis covariance matrix is reshaped to the known representation

$$B^{(a)} = (I - BH^T(R + HBH^T))^{-1}B. \tag{69}$$

The equivalence of both formulas is proven in Lemma 5.4.2 in Nakamura and Potthast [52] for example. With the help of the definition of B in Equation (33), the representation can further reformulated to

$$\begin{aligned} B^{(a)} &= (I - \gamma XX^T H^T (R + H\gamma XX^T H^T)^{-1} H)\gamma XX^T && (70) \\ &= \gamma X(I - \gamma Y^T (R + \gamma YY^T)^{-1} Y)X^T && (71) \\ &= \gamma X(I - Y^T(\gamma^{-1}R + YY^T)^{-1}Y)X^T. && (72) \end{aligned}$$

The application of equality Equation (51) in Equation (72) in combination with the definition of A (Equation (18)) leads to

$$\begin{aligned} B^{(a)} &= \gamma X(I - (\gamma^{-1}I + A)^{-1}A)X^T && (73) \\ &= \gamma X((\gamma^{-1}I + A)^{-1}(\gamma^{-1}I + A - A))X^T && (74) \\ &= X(\gamma^{-1}I + A)^{-1}X^T. && (75) \end{aligned}$$

so that the analysis covariance matrix in ensemble space is given by

$$B_{\text{ens}}^{(a)} := (\gamma^{-1}I + A)^{-1}. \tag{76}$$

The insertion of Equation (75) in the definition of $x^{(a,l)}$ in Equation (39) yields

$$\begin{aligned} x^{(a,l)} &= x^{(b,l)} + X(\gamma^{-1}I + A)^{-1}X^T H^T R^{-1}(y - Hx^{(b,l)}) && (77) \\ &= \bar{x}^{(b)} + x^{(b,l)} - \bar{x}^{(b)}X(\gamma^{-1}I + A)^{-1}Y^T R^{-1}(y - \bar{y}^{(b)} - Ye_l). && (78) \end{aligned}$$

The second step results from the application of Equation (45). The equation can be further reshaped with the equality $x^{(b,l)} - \bar{x}^{(b)} = Xe_l$ and the multiplication of $I = AA^{-1}$, i.e.,

$$\begin{aligned} x^{(a,l)} &= \bar{x}^{(b)} + X(e_l + (\gamma^{-1}I + A)^{-1}AA^{-1}Y^T R^{-1}(y - \bar{y}^{(b)} - Ye_l)) && (79) \\ &= \bar{x}^{(b)} + X(e_l + (\gamma^{-1}I + A)^{-1}A(C - e_l)). && (80) \end{aligned}$$

The last formulation results from the definition of the projected observation vector C given in Equation (47) and the definition of A in Equation (18). The ensemble representation of the analysis expectation is then given by

$$\beta^{(a,l)} := e_l + (\gamma^{-1}I + A)^{-1}A(C - e_l) \in \mathbb{R}^L. \tag{81}$$

Since the l-th unit vector $e_l \in \mathbb{R}^L$ denotes the l-th background particle in ensemble space, the second summand denotes the shift vectors, i.e.,

$$\beta^{(\text{shift},l)} := (\gamma^{-1}I + A)^{-1}A(C - e_l) \in \mathbb{R}^L. \tag{82}$$

All shift vectors are taken together in the matrix

$$W^{(\text{shift})} := \left(\beta^{(\text{shift},1)}, \ldots, \beta^{(\text{shift},L)}\right) \in \mathbb{R}^{L \times L}. \tag{83}$$

3.2.3. Draw Particles From Gaussian Mixture Distribution

In the last part of the LMCPF method the analysis ensemble is perturbed to increase the variability. To this end, new particles are drawn from a Gaussian distribution around each shifted particle which was previously selected. If a particle is selected multiple times, the same amount of particles is drawn from the respective Gaussian distribution. This approach equals the generation of L particles following the Gaussian mixture distribution in ensemble space, i.e.,

$$p_{\text{ens}}^{(a)}(\beta) := c_{\text{ens}}^{(a)} \sum_{l=1}^{L} e^{-\frac{1}{2}(\beta - \beta^{(a,l)})^T(\sigma(\rho)^2 B_{\text{ens}}^{(a)})^{-1}(\beta - \beta^{(a,l)})}, \qquad \beta \in \mathbb{R}^L. \tag{84}$$

The covariance matrix of each Gaussian is inflated by the factor $\sigma(\rho) \in \mathbb{R}_{>0}$ to control the ensemble spread. The variable ρ

denotes the inflation factor implemented in the LETKF method (see [4]), which follows an ansatz introduced by Desroziers et al. [55] and Li et al. [45]. Based on statistics of observations minus background

$$d^{o-b} = y - H\bar{x}^{(b)} \tag{85}$$

an adaptive inflation factor is calculated (see [55] or section e on page 352f. of Potthast et al. [31]), i.e.,

$$\tilde{\rho} = \frac{(d^{o-b})^T d^{o-b} - \text{trace}(R)}{\text{trace}(HBH^T)}. \tag{86}$$

To smooth the factor over time, the formula

$$\rho = \alpha\tilde{\rho} + (1-\alpha)\rho_{\text{old}} \tag{87}$$

is applied for some $\alpha \in [0, 1]$ and the inflation factor ρ_{old} of the previous time step. In the LMCPF method as well as the LAPF method, the inflation factor ρ of the LETKF method is scaled. The factor $\sigma(\rho)$ is derived by

$$\sigma(\rho) := \begin{cases} c_0, & \rho < \rho^{(0)}, \\ c_0 + (c_1 - c_0)\cdot \frac{\rho-\rho^{(0)}}{\rho^{(1)}-\rho^{(0)}}, & \rho^{(0)} \leq \rho \leq \rho^{(1)}, \\ c_1, & \rho > \rho^{(1)} \end{cases} \tag{88}$$

with parameters $\rho^{(0)}, \rho^{(1)} \in \mathbb{R}_+$ and $c_0, c_1 \in \mathbb{R}_+$. In the LETKF method, the analysis ensemble is inflated around the analysis ensemble mean. In the LAPF and LMCPF method, particles are resampled from the background ensemble, shifted (in case of the LMCPF) and then randomly perturbed to increase the ensemble variability. Due to these differences in the multiplicative inflation approach, the application of a scaled version of ρ is necessary and yielded better results in experiments. The boundaries c_0 and c_1 are tuning parameters. Due to the random drawing around each resampled particle, the parameters c_0 and c_1 should be chosen smaller than the parameters $\rho^{(0)}, \rho^{(1)}$ in the LETKF method. These parameters describe the upper and lower bound of ρ.

All in all, the steps of selecting, moving and drawing can be combined in the matrix W_{LMCPF}, i.e.,

$$W_{\text{LMCPF}} := \breve{W} + W^{(\text{shift})}\,\breve{W} + \sigma(\rho)\cdot[B^{(a)}_{\text{ens}}]^{1/2}\cdot Z. \tag{89}$$

with $\breve{W}$ defined in Equation (68), $W^{(\text{shift})}$ following Equation (83) and a random matrix $Z \in \mathbb{R}^{L\times L}$ with standard normally distributed random numbers as entries. Then, the full analysis ensemble is obtained by

$$(x^{(a,l)})_{l=1,\ldots,L} = \bar{x}^{(b)}\cdot \mathbb{1} + X\cdot W_{\text{LMCPF}} \tag{90}$$

where $\mathbb{1} \in \mathbb{R}^{1\times L}$ describes a row vector with ones as entries and X the ensemble pertubation matrix defined in Equation (11).

In Feng et al. [56], two nonlinear filters are compared which can preserve the first and second moments of the classical particle filter. First, the local particle filter in its version introduced in Poterjoy et al. [57] represent a localized adaption of the classical particle filter. Second, the local nonlinear ensemble transform filter (LNETF; [16]) is an approximation to the classical particle filter as well but instead of a classical resampling step a deterministic square root approach is followed. This is based on ideas of LETKF. Compared to the local particle filter and LNETF, the LMCPF uses a Gaussian mixture probability density function to approximate the background. With the stratified resampling step the particles are resampled following the posterior distribution, which is exact for Gaussian mixtures and Gaussian observation error. Due to the assumption of Gaussian mixture densities, the resampled particles are shifted which results in the exact mean vectors of the Gaussians of the posterior pdf, and also, temporarily, the exact covariances. To increase the variability of the ensemble, new particles are drawn from the posterior distribution as follows. Around each particle, new particles are randomly drawn from a Gaussian distribution with the exact mean vector and the exact covariance multiplied with an inflation factor. In contrast to the local particle filter, there is no rescaling of the ensemble applied in the LMCPF method. That means, the LMCPF will preserve the moments of a Gaussian mixture filter approximately up to sampling errors and inflation.

3.3. Localized Ensemble Transform Kalman Filter

The Localized Ensemble Transform Kalman Filter (LETKF) is first introduced in Hunt et al. [4] and is widely used in numerical weather prediction (e.g., [58]). The LETKF is based on equations of the Ensemble Kalman Filter (EnKF; [1, 3, 59]) transformed and performed in ensemble space. As the LAPF and LMCPF the observation error is chosen to be Gaussian distributed with the pdf described in Equation (10). In contrast to the methods described previously, this method assumes the background ensemble to represent a Gaussian distribution as well, i.e.,

$$p^{(b)}(x) := c^{(b)}\cdot e^{-\frac{1}{2}(x-\bar{x}^{(b)})^T G^{-1}(x-\bar{x}^{(b)})}, \quad x \in \mathbb{R}^n. \tag{91}$$

G denotes the estimated background covariance matrix following Equation (33) with $\gamma = 1/(L-1)$, i.e.,

$$G := \frac{1}{L-1}XX^T \in \mathbb{R}^{n\times n}. \tag{92}$$

To distinguish from the more general version of the covariance matrix introduced in Section 3.2 about the LMCPF method, the standard covariance estimator is named G. The transformed version in ensemble space—which is spanned by the columns of X in Equation (11)—is then given by

$$G_{\text{ens}} := \frac{1}{L-1}I_L \in \mathbb{R}^{L\times L} \tag{93}$$

with the $L \times L$ - identity matrix I_L. The application of Bayes' formula (9) to the background distribution $p^{(b)}$ and the observation error pdf Equation (10) leads to the Gaussian analysis pdf

$$p^{(a)}(x) = c^{(a)}e^{\left(-\frac{1}{2}(x-\bar{x}^{(a)})^T(G^{(a)})^{-1}(x-\bar{x}^{(a)})\right)} \tag{94}$$

with the covariance matrix

$$G^{(a)} = (G^{-1} + H^T R^{-1} H)^{-1} \tag{95}$$

and the expectation vector

$$\bar{x}^{(a)} = \bar{x}^{(b)} + G^{(a)} H^T R^{-1} (y - H\bar{x}^{(b)}). \tag{96}$$

The derivation can be found for example in Nakamura and Potthast [52] or in Evensen et al. [3]. A more common formulation of the update equations can be derived by rearrangement of Equations (95) and (96). Following Lemma 5.4.2 in Nakamura and Potthast [52], an equivalent form of the covariance matrix is given by

$$G^{(a)} = (I_n - GH^T(R + HGH^T))^{-1} G = (I_n - K)^{-1} G \tag{97}$$

with the Kalman gain matrix $K \in \mathbb{R}^{n\times n}$ and identity matrix $I_n \in \mathbb{R}^{n\times n}$. The covariance matrix in ensemble space is derived in Equations (70)–(98), i.e.,

$$G^{(a)}_{\text{ens}} := ((L-1)\cdot I_L + A)^{-1} \tag{98}$$

with identity matrix $I_L \in \mathbb{R}^{L\times L}$ and A defined in Equation (18). The insertion of Equation (75) applied to $G^{(a)}_{\text{ens}}$ in the definition of $\bar{x}^{(a)}$ in Equation (96) leads to

$$\begin{aligned}\bar{x}^{(a)} &= \bar{x}^{(b)} + X((L-1)\cdot I + A)^{-1} X^T H^T R^{-1} (y - H\bar{x}^{(b)}) &(99)\\ &= \bar{x}^{(b)} + X \cdot G^{(a)}_{\text{ens}} Y^T R^{-1} (y - \bar{y}^{(b)}). &(100)\end{aligned}$$

That means, the analysis mean in ensemble space is given by

$$\bar{\beta}^{(a)} := G^{(a)}_{\text{ens}} Y^T R^{-1} (y - \bar{y}^{(b)}) \in \mathbb{R}^L. \tag{101}$$

There are multiple approaches to obtain the full analysis ensemble in dependence on the analysis covariance matrix. The LETKF is based on the square root method. The weighting matrix W_{LETKF} is defined by the square root

$$W_{\text{LETKF}} = [(L-1)G^{(a)}_{\text{ens}}]^{\frac{1}{2}} \tag{102}$$

which is related to the covariance matrix by

$$G^{(a)}_{\text{ens}} = (L-1) W_{\text{LETKF}} (W_{\text{LETKF}})^T. \tag{103}$$

Additionally, the posterior covariance is inflated. To this end, an adaptive inflation factor ρ based on observation minus background statistics is derived by Equations (86) and (87). Then, the full analysis ensemble is calculated by

$$(x^{(a,l)})_{l=1,\dots,L} = \bar{x}^{(a)} \cdot \mathbb{1} + X \cdot \sqrt{\rho} \cdot W_{\text{LETKF}} \tag{104}$$

where $\mathbb{1} \in \mathbb{R}^{1\times L}$ describes a row vector with ones as entries and X the ensemble pertubation matrix defined in Equation (11).

4. STUDY OF INDIVIDUAL STEPS OF LMCPF

The LMCPF method can be divided in three parts: stratified resampling (Section 3.2.1), shift of particles (Section 3.2.2) and drawing new particles from a Gaussian mixture distribution (Section 3.2.3). In this section, we discuss the behavior of the ensemble during the different parts of a single data assimilation step performed by the LMCPF method.

4.1. Stratified Resampling

The stratified resampling step represents the main idea of the particle filter method. Only the particles with sufficient weight are chosen. In the LAPF and LMCPF methods, the resampling step is carried out in the ensemble space in order to reduce the dimension and prevent filter collapse. This step occurs in both methods but different particle weights are used. The relative weights of the LAPF Equation (26) depend on the distance of the particles to the observation and the observation error covariance. In case of the LMCPF, the exact weights Equation (66) additionally depend on the particle uncertainty parameter κ.

Figure 2 illustrates the relation between these two weights. If κ tends to zero, the normalized Gaussian mixture weights tend to the classical particle filter weights, which are used in the LAPF and were previously used in the LMCPF method. The particle weights are derived from the case illustrated in **Figure 3**. The approximate weights in **Figure 2** suggest that in the LAPF method only one particle would have been chosen as one particle gets all the weight. Furthermore, the exact weights approach each other for larger κ. That means, more particles would be chosen in the stratified resampling step for larger κ. If κ tends to infinity, the exact weights tend to one so that the probability to sample a particle is the same for each particle.

Since the relative weights depend on the distance of the particles to the observation, these background particles, which are close to the observation, are chosen. This is illustrated in **Figure 3** as well as in the example with a bimodal background distribution in **Figure 4**. In the bimodal case, all the particles of the mode close to the observation are resampled. In both examples, the observation is located outside of the background ensemble. After the stratified resampling step, the particles are still far from the observation. In **Figure 4B**, the shifted ensemble mean of the LETKF method is even closer to the observation than the nearest background particles. That leads to the idea, to use a Gaussian mixture representation in the LMCPF, to include the shifting step of the LETKF, which is discussed in the next part.

4.2. Shift of Particles

In contrast to the ensemble Kalman filter method, particle filters do not shift particles toward the observation but only choose the nearest ones, so that the ensemble mean is pulled toward the observation. In the LMCPF, each remaining particle is shifted as the ensemble mean in the ensemble Kalman filter method. Furthermore, the shift is affected by the particle uncertainty described by the background covariance matrix. Modification of the parameter κ in Equation (34) yields changes in the valuation of the particle uncertainty. If κ is set to a larger value, there is less

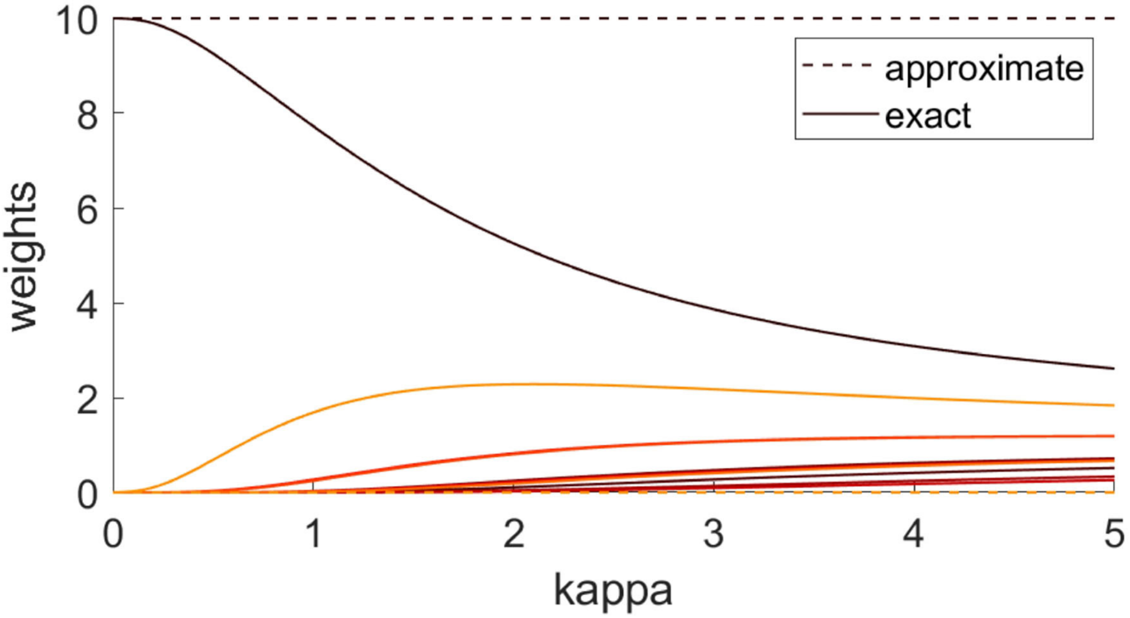


FIGURE 2 | The exact Gaussian mixture weights $w^{(a,l)}$ Equation (66) are compared against the approximate weights $\tilde{w}^{(a,l)}$ Equation (26), which are used in the LAPF method. Each color denotes the pair of weights (approximate and exact) for one of the 10 particles. The particle weights come from the scenario illustrated in **Figure 3**. For the exact weights, the particle uncertainty parameter κ is varied.

confidence in the background ensemble. Hence, the confidence in the observation ascends, relatively seen. Finally, this results in a stronger shift of the remaining particles toward the observation. To validate this intuition mathematically, the spectral norm of the posterior covariance matrix

$$B_{\text{ens}}^{(a)} = \left(\frac{L-1}{\kappa} I_L + A \right)^{-1} \tag{105}$$

with $\kappa > 0$, the identity matrix $I_L \in \mathbb{R}^{L \times L}$ and projected observation error covariance matrix

$$A = Y^T R^{-1} Y \in \mathbb{R}^{L \times L} \tag{106}$$

is observed. The spectral norm is induced from the euclidean vector norm and is defined by the square root of the maximal eigenvalue of $A^T A$. In the case of complex matrices, the transpose matrix is replaced by the adjoint matrix. Matrix A is symmetric as the observation error covariance matrix R is a symmetric matrix by definition. Furthermore, every symmetric matrix is normal. Let be $U \in \mathbb{R}^{L \times L}$ the matrix with eigenvectors of the normal matrix A as columns and $D \in \mathbb{R}^{L \times L}$ the diagonal matrix with the respective eigenvalues as diagonal entries ordered from maximal to minimal eigenvalue such that

$$A = UDU^T \tag{107}$$

holds. Since U is a unitary matrix, i.e., $UU^T = I_L$, the inverse term of $B_{\text{ens}}^{(a)}$ can be reshaped to

$$\frac{L-1}{\kappa} I_L + UDU^T = U \left(\frac{L-1}{\kappa} I_L + D \right) U^T. \tag{108}$$

That means, U also describes the unitary matrix of the eigenvalue decomposition of the inverse of $B_{\text{ens}}^{(a)}$ and the eigenvalues are given by

$$\lambda_i = \frac{L-1}{\kappa} + \mu_i, \qquad i = 1, \dots, L \tag{109}$$

with eigenvalues $(\mu_i)_i$ of A. We remark that $\mu_i > 0$ holds for all $i = 1, \dots, L$ as A is positive definite. The spectral norm of the inverse matrix equals the inverse of the smallest eigenvalue $\min\{\lambda_i | i = 1, \dots, L\}$, i.e.,

$$\|B_{\text{ens}}^{(a)}\|_2 = \left(\frac{L-1}{\kappa} + \min_{i=1,\dots,L} (\mu_i) \right)^{-1}. \tag{110}$$

On the basis of this term, we can easily see that larger values for κ leads to a larger spectral norm of $B^{(a)}$.

Furthermore, the shift vectors are defined by

$$\beta^{(\text{shift},l)} = \left(\frac{L-1}{\kappa} I_L + A \right)^{-1} A(C - e_l), \qquad l = 1, \dots, L. \tag{111}$$

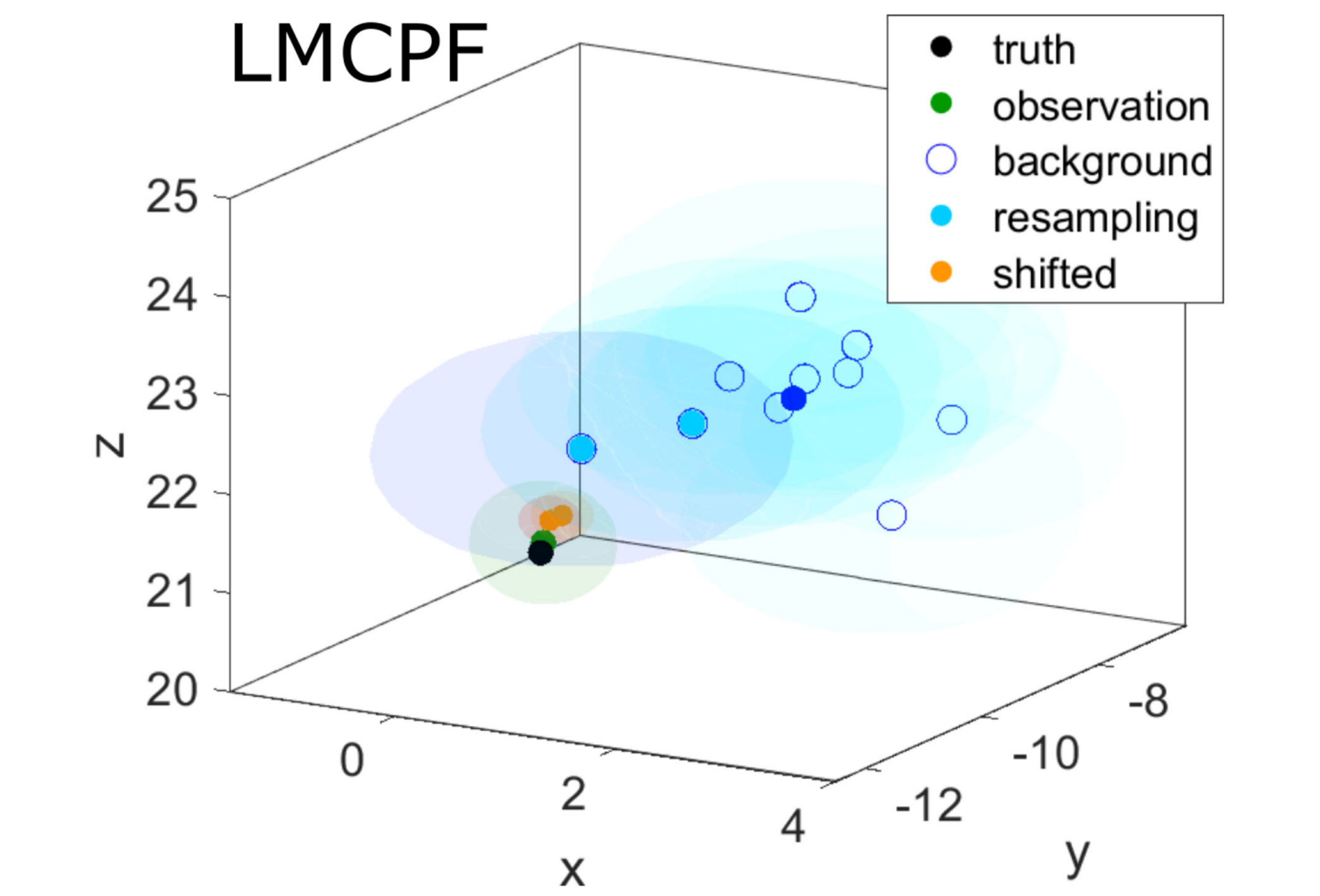


FIGURE 3 | A single assimilation step is carried out with the LMCPF method. The observation (green point) is located outside of the background ensemble of size $L = 10$ with the ensemble mean represented by the dark blue point. The particles chosen in the stratified resampling step (light blue points) are shifted toward the observation (orange points). The particle uncertainty parameter κ is set to one. The shaded areas denote Gaussian ellipsoids with respect to the corresponding covariance matrices. Darker colored ellipsoids around the background particles denote larger weights $w^{(a,l)}$ defined in Equation (66).

To discover the shifting strength for different κ, the spectral norm of $B_{\text{ens}}^{(a)}$ multiplied with A is examined. With the eigenvalue decomposition of A, we obtain

$$\left(\frac{L-1}{\kappa}I_L + UDU^T\right)^{-1} UDU^T$$

$$= (U^T)^{-1}\left(\frac{L-1}{\kappa}I_L + D\right)^{-1} U^{-1}UDU^T \qquad (112)$$

$$= U\left(\frac{L-1}{\kappa}I_L + D\right)^{-1} DU^T \qquad (113)$$

which follows from the property $U^{-1} = U^T$ of a unitary matrix U. This results in the spectral norm

$$\|B_{\text{ens}}^{(a)}A\|_2 = \max_{i=1,\ldots,L}\left\{\left(\frac{L-1}{\kappa} + \mu_i\right)^{-1} \mu_i\right\} \qquad (114)$$

which gets larger for greater κ.

In **Figure 3**, the shift of the two particles, which are chosen in the stratified resampling step results in particles close to the observation even for $\kappa = 1$. For this parameter choice, the background error covariance matrix B equals the standard covariance estimator. The shaded areas around the dots describe the uncertainty. Compared to the background uncertainty, the observation error covariance matrix $R = 0.3^2 \cdot I$ is smaller, which explains the strong shift toward the observation. In comparison, the difference between background and observation uncertainty is smaller in the bimodal case in **Figure 4**. This results in shifted particles, which are not as close to the observation as in **Figure 3**.

4.3. Draw Particles From Gaussian Mixture Distribution

In the LMCPF as well as in the LAPF method, new particles are drawn from a Gaussian mixture distribution but different covariance matrices are applied. In the LAPF, an inflated version of the background error covariance matrix in ensemble space $1/(L-1) \cdot I$ is used. The covariance matrix is adapted by the spread control factor $\sigma(\rho)^2$, which is derived in Equation (88). In contrast, the newly derived covariance matrix $B_{\text{ens}}^{(a)}$ Equation (98) in ensemble space is applied in an inflated version in the LMCPF.

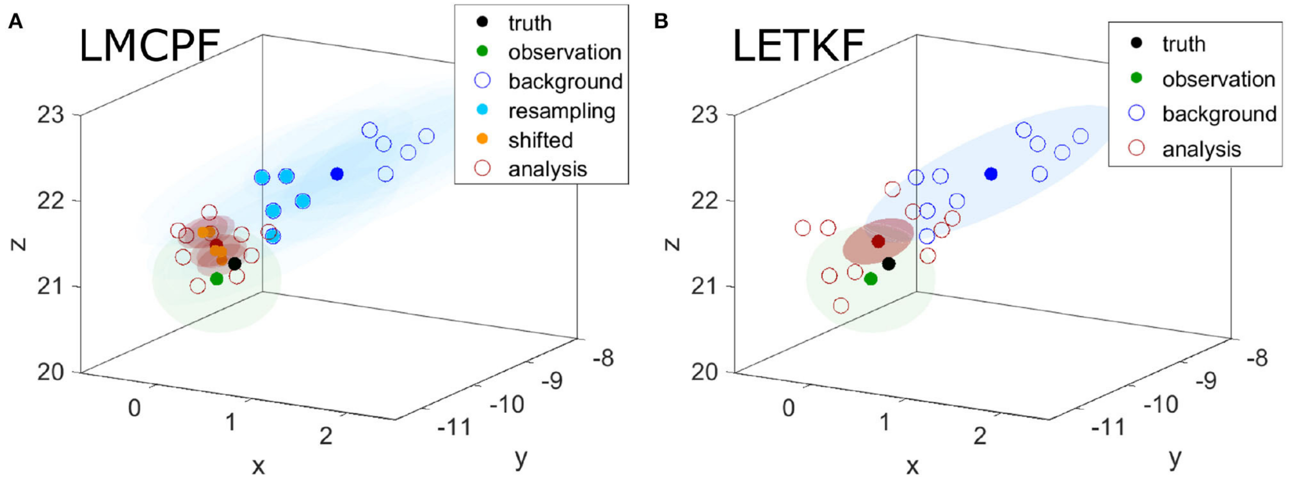


FIGURE 4 | The background ensemble (blue circles) is generated from a bimodal distribution and the observation (green point) is located near one of the modes. The dark blue point illustrates the background ensemble mean. In **(A)**, the assimilation step is performed with the LMCPF method and in **(B)** with the LETKF method. The light blue points in **(A)** illustrate the resampled particles and the orange points describe the shifted particles for $\kappa = 1$. The analysis particles resulting from LMCPF and LETKF are represented by the red circles and the red point illustrates the analysis ensemble mean. In the LMCPF method, these particles are randomly generated from Gaussian distributions with the shifted particles as expectation vectors. The shaded areas denote Gaussian ellipsoids with respect to the corresponding covariance matrices.

TABLE 1 | Parameters of the model configuration and experimental setup for the Lorenz 1963 (L63) and 1996 (L96) models.

	Forecast length Δ_t	Model param.	Std of obs error σ_{obs}	Obs. variables	DA steps
L63	0.15; 0.3; 0.5	$\sigma^{\text{true}} = 10; \sigma = 12$	0.5	First	1, 000
L96	0.3; 0.5	$F^{\text{true}} = 8; F = 8:9.5$	0.2; 0.5; 0.8; 1.1	Every second	1, 000

The draw from a Gaussian mixture distribution is carried out by drawing new particles from Gaussian distributions around each chosen particle. For all Gaussian distributions, the same covariance matrix is applied. In case of the LMCPF, the spectral norm of the covariance matrix $B_{\text{ens}}^{(a)}$ results in a larger value if the particle uncertainty parameter κ is set to a greater value. This counteracts the effect that a stronger shift toward the observation vector leads to smaller distances among the particles.

Figure 4 shows the results of one LMCPF and LETKF step for a bimodal background distribution. The Gaussian ellipsoids cover random draws from the same three dimensional distribution with a high probability. Nevertheless, the analysis particles of LMCPF and LETKF are located outside of the ellipsoids. The particles are resampled in the $L-1$-dimensional ensemble space and not in the three-dimensional model space. This leads to a wider analysis ensemble for $L > n$ than we would obtain by drawing in the n-dimensional model space. In practice, the dimension of the model space is much larger than the dimension of the ensemble space so that this case does not occur.

In comparison to the particle filter method, the analysis ensemble derived by the LETKF method maintains the structure of the background ensemble and is only shifted and contracted. In that case, the ensemble mean, which represents the state estimate, is not located in an area with high probability density but in between the two modes (see **Figure 4B**). The analysis ensemble aims to approximate the uncertainty distribution of the state estimate. This more realistic uncertainty estimation is one of the advantages of the particle filter methods over the ensemble Kalman filter.

5. RESULTS FOR LONGER ASSIMILATION PERIODS

In the following, the results of longer data assimilation experiments for the Lorenz 1963 model as well as the 40-dimensional Lorenz 1996 model are discussed. Beside the comparison of root-mean-square errors following Equations (115) and (116) for different methods, the development of the effective ensemble size [see Equations (119) and (120)] in the particle filter methods are observed. For both models, the initial ensemble size is set to $L = 20$ in the following experiments. Further parameters of the model configuration and experimental setup, which are used in this section, are summarized in **Table 1**.

For the 40-dimensional Lorenz 1996 model, the methods are used in a localized form, as described at the beginning of Section 3. The localization depends on the localization radius r_{loc}, which affects the number of observations used in the analysis step. Moreover, the optimal localization radius depends on the method as well as the model parameters. For the LETKF method, we choose r_{loc} in between 4 and 7 in depending on the model error, the integration time Δ_t and the observation noise after the investigation of different localization radii. With respect to the LMCPF with exact weights, the localization radius r_{loc} is set to a value between 4 and 6 in the experiments of this section. In addition, experiments revealed larger effective ensemble sizes for smaller localization radii. Moreover, an automatic restart was introduced for all methods to catch extreme cases.

5.1. Definition of RMSE and Effective Ensemble Size

To compare different data assimilation methods, a measure is needed. In general, the goodness of a DA method is associated with the distance between the background or analysis state estimate and the truth, or alternatively the observation if the truth is not available. For that purpose, the normalized euclidean norm or root-mean-square-error (RMSE) is used to calculate the distance of background or analysis state estimate and the truth at time t_k, i.e.,

$$e_k^{(b)} : = e^{(b)}(t_k) = \frac{1}{\sqrt{n}} \left\| \bar{x}_k^{(b)} - x_k^{true} \right\|_2 , \tag{115}$$

$$e_k^{(a)} : = e^{(a)}(t_k) = \frac{1}{\sqrt{n}} \left\| \bar{x}_k^{(a)} - x_k^{true} \right\|_2 , \tag{116}$$

where $n \in \mathbb{N}$ denotes the number of variables of the underlying model and $\bar{x}_k^{(b)}, \bar{x}_k^{(a)}$ describe the background or analysis ensemble means. For a time period given, where data assimilation is carried out at the measurement points $t_1, \ldots, t_K$, the averaged errors are denoted by

$$e^{(b)} = \frac{1}{K} \sum_{k=1}^{K} e_k^{(b)}, \tag{117}$$

$$e^{(a)} = \frac{1}{K} \sum_{k=1}^{K} e_k^{(a)}. \tag{118}$$

In terms of particle filter methods, the development of the effective ensemble size is an important quantity to examine the stability of the filter. The effective ensemble size is defined by

$$L_{\text{eff}} = \frac{1}{\sum_{l=1}^{L} (w^{(a,l)}/L)^2} \tag{119}$$

with the relative particle weights in ensemble space $w^{(a,l)}$ of the LMCPF described in Equation (66) or with the classical particle filter weights $\tilde{w}^{(a,l)}$ of the LAPF defined in Equation (26). In general, particle filter methods suffer in high-dimensional spaces from filter degeneracy due to the finite ensemble size (see [6]). In that case, the effective ensemble size tends to one, which means that the weights become strongly non-uniform. With respect to the 40-dimensional Lorenz 1996 model, the effective ensemble size is computed at each localization point and the average at each data assimilation cycle is derived. The mean effective ensemble size over all localization points is denoted by

$$\bar{L}_{\text{eff}} = \frac{1}{P} \sum_{p} L_{\text{eff}}(p) \tag{120}$$

where P describes the number of localization points ($P = n$ for Lorenz 1996) and L_{eff} is calculated at each localization point using the respective weights.

5.2. LMCPF Results in Dependence of the Particle Uncertainty Parameter κ

The results of data assimilation methods vary in dependence of the model parameters integration time Δ_t of the dynamical model, the model error between true and model run and observation noise σ_{obs}. The chaotic behavior of the Lorenz systems means that small differences in the initial conditions can lead to significantly different future trajectories. In average, greater propagation or forecast time intervals result in greater perturbations of the model run. The nonlinearity of the Lorenz models causes the propagation of some Gaussian distributed ensemble to result in non-Gaussian structures even at shorter lead times.

Figure 5 shows the integration of a Gaussian distributed ensemble over time with Lorenz 1963 model dynamics. For $\Delta_t = 0.3$ and $\Delta_t = 0.5$, the resulting ensemble is clearly non-Gaussian so that the main assumption of the Kalman filter to the background distribution does not hold. As a consequence, we expect improvements of LMCPF over LETKF especially for longer forecast times.

Moreover, model error means that true states, respectively, observations are generated by a slightly different dynamical model than the first guess from the previous analysis ensemble. For the Lorenz systems, the model error is produced by the application of different values for the Prandtl number σ (Lorenz 1963) and for the forcing term F (Lorenz 1996). In NWP systems, the atmospheric model is known to have errors. Hence, it is important to investigate the application of data assimilation methods in case of model error. Naturally, we expect the model run to differ stronger from the true run for greater differences in the model parameters.

In addition, the observation noise σ_{obs} strongly affects the data assimilation results. As in the case of the model error, this is no surprise, since the observation is used in data assimilation to obtain an analysis state. The LMCPF is quite sensitive to the observation noise because the resampling as well as the shift moves the ensemble toward the observation. To generate the observations for experiments with the Lorenz models, the true trajectory is randomly perturbed at time points, where data assimilation is performed. If some observation is far from the truth by chance, an overestimation of the importance of this observation might lead to worse results of the LMCPF compared to LETKF or LAPF.

There are six parameters in the LMCPF method to adapt the method to model and observation error as well as the integration time. The five parameters ρ_0, ρ_1, c_0, c_1 and α are used to control the spread of the analysis ensemble in the last step, where new particles are drawn from a Gaussian mixture distribution (see Section 3.2.3) . But the sixth, the particle uncertainty parameter κ, respectively, γ defined in Equation (34), is the most important parameter since the variable affects the spread of the analysis ensemble as well as the movement of the particles toward the observation.

In the following, the results for LMCPF compared to LETKF are shown for different settings of Lorenz 1963 and 1996. To identify a reasonable particle uncertainty parameter κ, the

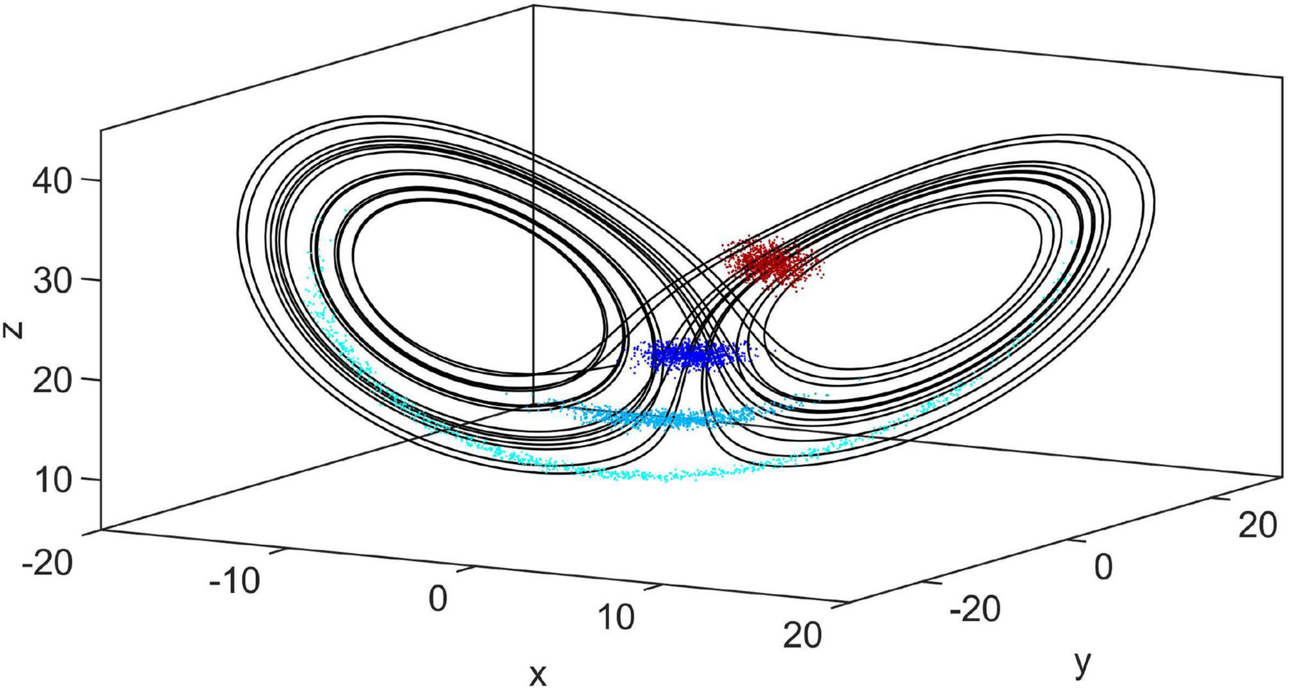


FIGURE 5 | A thousand particles drawn from a Gaussian distribution (red points) are integrated in time with respect to the Lorenz 1963 model dynamics for $\Delta_t = 0.15$ (blue), $\Delta_t = 0.3$ (lightblue) and $\Delta_t = 0.5$ (cyan) time units.

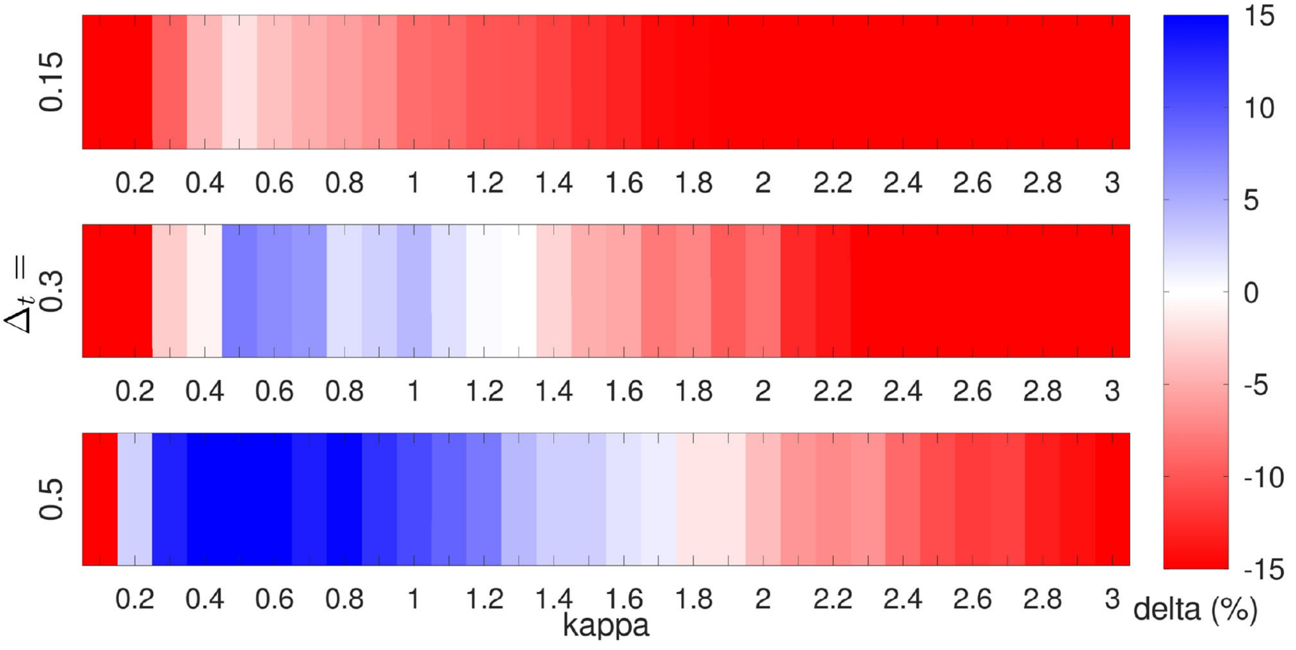


FIGURE 6 | Comparison of background errors of LMCPF and LETKF following Equation (121) for different forecast lengths $\Delta_t = 0.15$, $\Delta_t = 0.3$ and $\Delta_t = 0.5$. Positive values denote a smaller RMSE of truth minus background for the LMCPF method than the LETKF. For each parameter combination, 1,000 data assimilation steps are carried out for the Lorenz 1963 model whereas the last 900 steps are used to compute the statistics. The experiments are repeated 10 times with different seeds and the average error is reported. The true trajectory is generated with $\sigma^{\text{true}} = 10$, the integration of the ensemble of states is performed with $\sigma = 12$ and the observation noise equals $\sigma_{\text{obs}} = 0.5$. Only the first variable is observed. The ensemble size is set to $L = 20$ for both methods.

parameter is varied. In **Figure 6**, experiments for different forecast lengths Δ_t are performed with respect to the Lorenz 1963 model. The observation error standard deviation is chosen as $\sigma_{\text{obs}} = 0.5$ and only the first variable is observed. The true trajectory is generated with the Prandtl number $\sigma^{\text{true}} = 10$, while the forecast ensemble is integrated with $\sigma = 12$ to introduce model error. For each parameter setting, 1,000 data assimilation cycles are carried out with both methods, whereas the average errors over the last 900 cycles are computed. That means the first 100 steps are not used. Furthermore, each experiment is repeated ten times with different seeds to generate different random numbers and the average error is reported. The mean background errors [see Equations (117) and (118)] of both methods are compared by

$$\delta = \frac{e^{(b)}_{\text{LETKF}} - e^{(b)}_{\text{LMCPF}}}{e^{(b)}_{\text{LETKF}}} \cdot 100. \tag{121}$$

Positive values (blue arrays) for δ denote better results for LMCPF than LETKF. Following **Figure 5**, the background ensemble is less Gaussian distributed for longer forecast lengths. **Figure 6** illustrates the improvement of LMCPF over LETKF in particular for $\Delta_t = 0.5$. In case of $\Delta_t = 0.15$, the results for LMCPF are worse than for LETKF. For a longer forecast length, the RMSE of background minus truth is lower than the RMSE of LETKF for a wider range of values for κ.

In **Figure 7**, the results for a range of values of κ are shown for the 40-dimensional Lorenz 1996 model with respect to different model errors. Similar to **Figure 6**, the background errors of LMCPF and LETKF are compared by Equation (121). One thousand data assimilation cycles are carried out, whereas the first 100 steps are considered as spin-up time and are not used in the computation of the mean errors. Moreover, the experiments are repeated ten times with different random seeds. To receive the results displayed in **Figure 7**, the truth is generated with the forcing term $F^{\text{true}} = 8$, while the forecast ensemble is derived with different forcing terms between $F = 8$ and $F = 9.5$. In addition, the observation error standard deviation is set to $\sigma_{\text{obs}} = 0.5$ and a longer forecast length $\Delta_t = 0.5$ is applied. The results indicate, that in most cases there is some particle uncertainty parameter κ, so that the LMCPF outperforms the LETKF.

Following Lei and Bickel [60], longer forecast lengths ($\Delta_t > 0.4$) lead to highly non-Gaussian ensembles for the 40-dimensional Lorenz 1996 model with forcing term $F = 8$. To verify this, we integrated a standard Gaussian distributed ensemble ($L = 10,000$) in time for $\Delta_t = 0.5$ and with forcing term $F = 8$. The distance of the resulting distribution to a Gaussian distribution with the same mean and variance can be measured by the distance of the skewness and kurtosis to the characteristic values 0 and 3 for skewness and kurtosis of a Gaussian distribution. For the integrated ensemble, we obtain 0.56 as absolute skewness averaged over all $N = 40$ model variables. The averaged absolute distance of the empirical kurtosis of the integrated ensemble to the characteristic value 3 of a Gaussian distribution is 0.99. This indicates a non-Gaussian ensemble.

An increasing value of F up to 9.5 leads to a larger distance of the background to the true state or the observations which denotes a larger systematic model bias. **Figure 7** illustrates that for larger model error, the RMSE of LMCPF is lower than for LETKF for a wider range of values for κ. That means, the parameter adjustment of the LMCPF is easier for larger model error. In case of no model error for the forcing term $F = 8$, the distance between observations and background is smaller than in cases with model error. In theory, we suggest that smaller values for the particle uncertainty parameter κ yield better results in that case since this leads to less uncertainty in the background. If κ tends to zero, the LMCPF gets more similar to LAPF. For the LAPF, we have observed a greater sensitivity to sampling errors. To this end, experiments for increased ensemble size ($L = 100$) were performed which showed better scores of LMCPF than LETKF in case of no model error and for smaller values of κ. Finally, the perfect model scenario with small distances between background and observation is a difficult case for the LMCPF with small ensemble sizes while this case is less relevant for the application in real NWP systems. In realistic applications, model errors occur and the applicable ensemble size is relatively small compared to the model dimension.

Furthermore, the effective ensemble size depends on the parameter κ. If κ tends to infinity, the effective ensemble size tends to the upper boundary L. This can be explained by **Figure 2**, which illustrates that the particle weights approach each other if κ tends to infinity. This means, that all the particles get the same weight, which results in the effective ensemble size $L_{\text{eff}} = L$. With respect to the experiments in **Figure 7**, the mean effective ensemble size varies for $\kappa > 0.5$ between $L_{\text{eff}} = 8$ and $L_{\text{eff}} = 15$. The variabilty of the effective ensemble size for different model errors is negligible. As remark, further experiments with different localization schemes and localization radii have shown that smaller localization radii lead to larger effective ensemble sizes up to a certain point. To ensure that the ability of the LMCPF to outperform the LETKF (see **Figure 7**) do not depend solely on the special selection on forcing terms F^{true} and F, additional combinations between 6.5 and 9.5 were tested.

In **Figures 6**, **7**, the results for different integration times and model errors are shown. **Figure 8** illustrates the changes for different observation standard deviations σ_{obs}. On the one hand, the LMCPF is able to outperform the LETKF for a wider range of values for κ. On the other hand, there is the tendency that for larger observation standard deviation smaller values for κ lead to good results. As the parameter κ adapts the particle uncertainty, smaller values decrease the uncertainty of the background ensemble and relatively increase the uncertainty of the observation. That means the particles are pulled less strongly in the direction of the observation.

In addition, we compared LMCPF and LETKF in case of non-Gaussian distributed observations. To this end, observations are generated with errors following a univariate non-Gaussian double exponential Laplace distribution [16], which are also applied in [56], and an equivalent experiment to **Figure 7** was performed. The observation error standard deviation is chosen as $\sigma_{\text{obs}} = 0.5$ again. There is no significant improvement of LMCPF compared to LETKF in case of non-Gaussian observations. Since

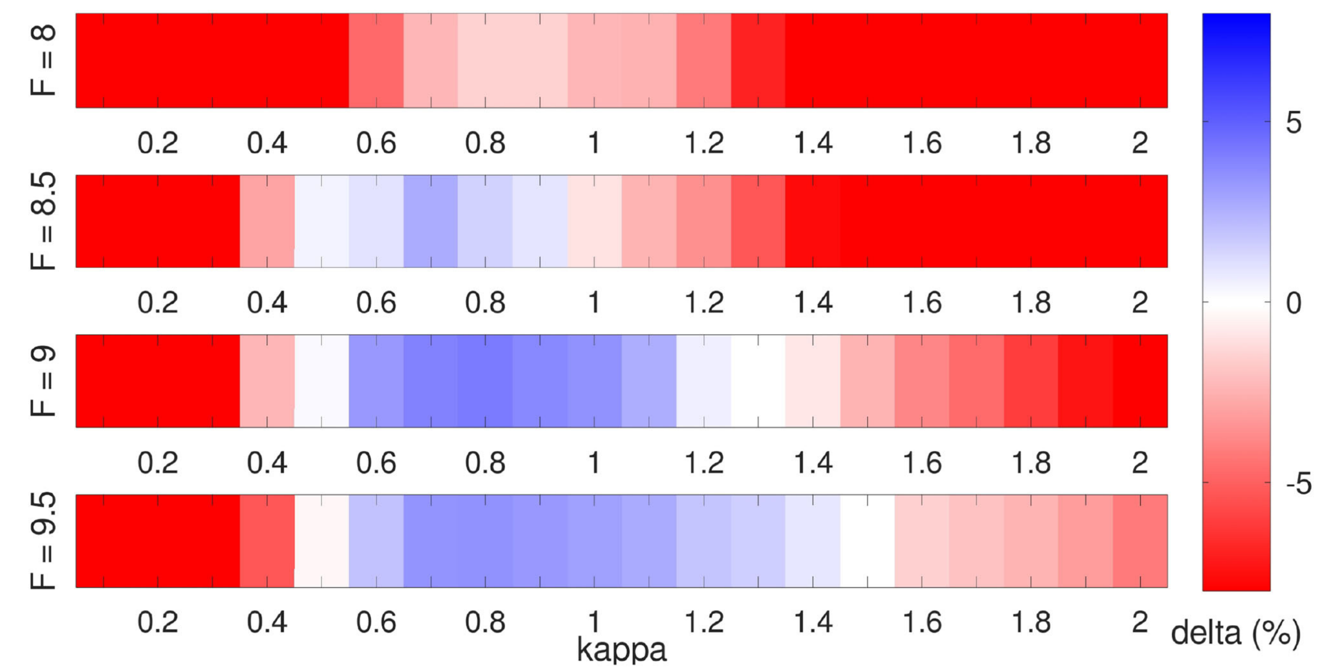


FIGURE 7 | Comparison of background errors of LMCPF and LETKF following Equation (121). Positive values denote a smaller RMSE of truth minus background for the LMCPF method than the LETKF. For each parameter combination, 1, 000 data assimilation steps for the Lorenz 1996 model are carried out whereas the last 900 steps are used to compute the statistics. The experiments are repeated ten times with different seeds and the average error is reported. The true trajectory is generated with $F^{\text{true}} = 8$, the forecast length is set to $\Delta_t = 0.5$ and the observation noise equals $\sigma_{\text{obs}} = 0.5$. Every second variable is observed. The ensemble size is set to $L = 20$ for both methods.

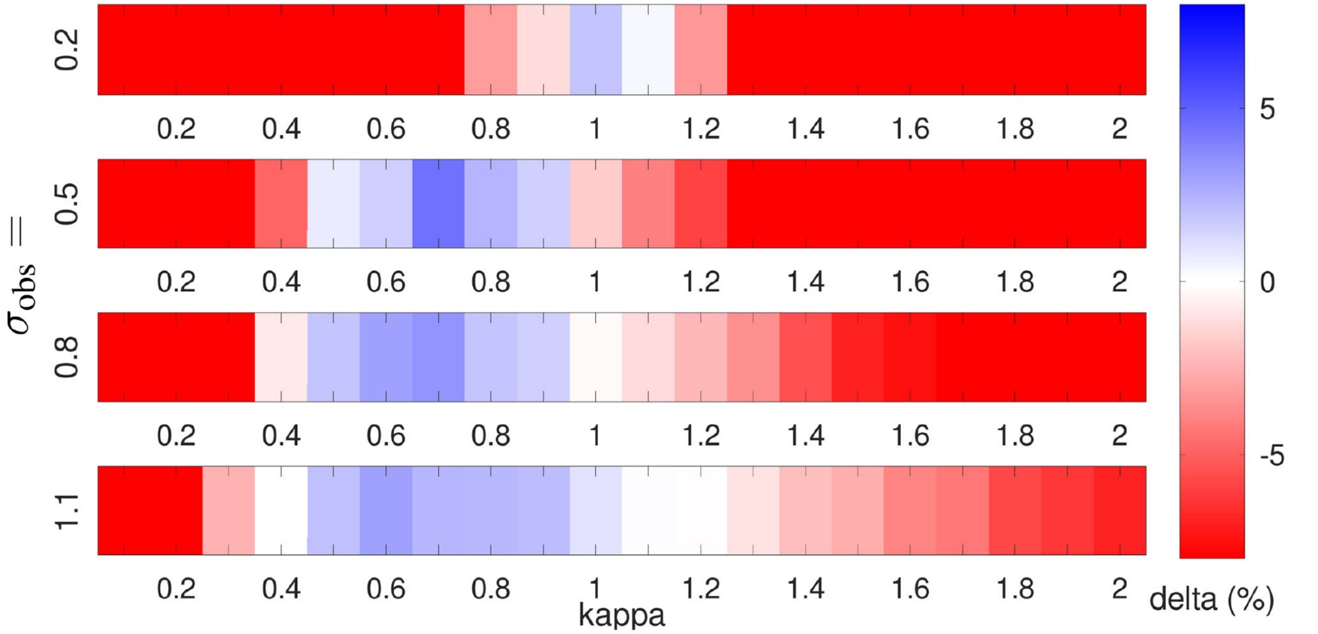


FIGURE 8 | The background errors of LMCPF and LETKF following Equation (121) are compared for different observation error standard deviations σ_{obs}. Positive values denote a smaller RMSE of truth minus background for the LMCPF method than the LETKF. For each parameter combination, 1, 000 data assimilation steps for the Lorenz 1996 model are carried out whereas the last 900 steps are used to compute the statistics. The experiments are repeated ten times with different seeds and the average error is reported. The true trajectory is generated with $F^{\text{true}} = 8$ and the integration of the ensemble of states is performed with $F = 8.5$. The forecast length is set to $\Delta_t = 0.5$. Every second variable is observed. The ensemble size is set to $L = 20$ for both methods.

both methods assume Gaussian distributed observation errors by definition, the results confirmed the expectation, that LMCPF does not have an advantage over LETKF in case of non-Gaussian observations. But there is the possibility to adapt the LMCPF in future to account for non-Gaussian observation error. Similar to the idea of a Gaussian mixture filter, the observation error distribution may be approximated by a sum of Gaussians. This would lead to new particle weights and shift vectors.

5.3. LMCPF With Gaussian Mixture and Approximate Weights

In the first version of the LMCPF method presented in Walter et al. [32], the particle weights are approximated by the classical particle filter weights in ensemble space, which are used in the LAPF method. This is reasonable if the covariance B of the Gaussians kernels is small compared to the distance of observation minus background particles. But this assumption may not be justified in practice. If the uncertainty parameter κ tend to zero the assumption is fulfilled and the exact Gaussian mixture weights tend to the approximate weights (see **Figure 2**).

In **Figure 9**, the LMCPF method with exact Gaussian mixture weights [see Equation (66)] is compared to the LMCPF method with approximate weights [see Equation (26)] in the case that every second variable is observed. To compare the methods for a variety of model parameters, the forecast length is set to $\Delta_t = 0.3$ for the experiments in the following sections. The results of LMCPF with exact and approximate weights are comparable but the overall background and analysis errors are higher for the version with approximate weights. Moreover, the adaptive inflation parameters ρ_0, ρ_1, c_0, c_1 and α are set to the same values for both methods and both methods have a similar ensemble spread averaged over the whole experiment. Furthermore, the ensemble spread is overestimated for both methods compared to the background, respectively, analysis error.

In **Figure 10**, the development of the effective ensemble size L_{eff} over the last 200 assimilation steps of this experiment is plotted for the LMCPF with exact and approximate weights as well as the LAPF method. The effective ensemble size of the LMCPF with approximate weights is only slightly higher than for the LAPF method, while the line of LMCPF with exact weights is significantly higher. Also, the localization radius has a large effect on the effective ensemble size. Smaller localization radii r_{loc} lead to larger effective ensemble sizes. Regarding the results in **Figure 10**, for the LMCPF method with exact weights, the localization radius is set to $r_{\text{loc}} = 4$, while for the other two methods, the radius is chosen as $r_{\text{loc}} = 2$. That means, for the same localization radius the effective ensemble size of LMPCF with exact weights would be even larger. Moreover, the localization radius is an important parameter to achieve stable results in case of the LAPF method. For the LMCPF method, the application of the exact Gaussian mixture weights lead to higher effective ensemble sizes so that the filter performance does not depend so heavily on the localization radius and optimal results are obtained for higher localization radii than for the version with approximate weights. Further experiments for longer forecast lengths ($\Delta_t = 0.5$ and $\Delta_t = 0.8$) have also shown that the effective ensemble size decreases for increasing integration time for all three particle filter versions. While the effective ensemble size of the LMCPF with exact weights still take values around $L_{\text{eff}} = 10$ for an initial ensemble size of $L = 20$, the variable decreases to values around $L_{\text{eff}} = 3$ for LAPF and LMCPF with approximate weights. The increase of the effective ensemble size shows the improvement of the stability of the LMCPF method with exact particle weights. In case of a larger effective ensemble size, more information of the background ensemble is used. If only few particles are chosen in the stratified resampling step, the ensemble spread depends more on the adaptive spread control parameters ρ_0, ρ_1, c_0, c_1 and α. In a worst case scenario where only one particle is chosen, all analysis particles are drawn from the same Gaussian distribution with inflated covariance matrix. Small changes in the covariance matrix of the Gaussian distribution effect the ensemble spread stronger compared to drawing the analysis particles from Gaussians with different expectation vectors. Using the exact Gaussian mixture weights, Kotsuki et al. [39] also detected an improvement of the stability of the LMCPF method with respect to the inflation parameters within an intermediate AGCM. Nevertheless, the application of the analysis covariance matrix $B_{\text{ens}}^{(a)}$ [see Equation (98)] in the Gaussian mixture distribution, from which new particles are drawn in the last step, leads for both LMCPF versions to more stable results with respect to the spread control parameters compared to the LAPF method.

5.4. Comparison of LMCPF, LAPF, and LETKF

In this section, the three localized methods LMCPF, LAPF and LETKF are compared with respect to the 40-dimensional Lorenz 1996 model.

Figures 11, **12** describe the results for the true forcing term $F^{\text{true}} = 8$ and $F = 9$ for the model integration with integration time $\Delta_t = 0.3$. Compared to the overall results in **Figure 9** for an experiment with larger model error $F = 9.5$, the RMSE of background or analysis mean minus truth for the LMCPF method takes lower values. Furthermore, the results for the last 200 data assimilation steps of the experiment in **Figure 11** illustrate that the higher errors for the LAPF method mostly come from high peaks at some points, while the errors are comparable for most regions. The tuning of the spread control parameters is essential to obtain good results for the LAPF. Compared to the LMCPF, the filter is more sensitive to these parameters. Additionally, background and analysis errors of the LMCPF method are lower than the errors of the LETKF and the LAPF methods for the majority of the shown time steps. The mean errors over the whole period except a spin-up phase, take lower values even if there are high peaks at some steps. Some outliers occur for each of the three methods.

The RMSE development gives an impression for the overall performance of the filters. In contrast, **Figure 12** illustrates the behavior for individual variables over the full period except a spin-up phase of 100 data assimilation steps. The difference between the background (**Figure 12A**) and analysis (**Figure 12B**)

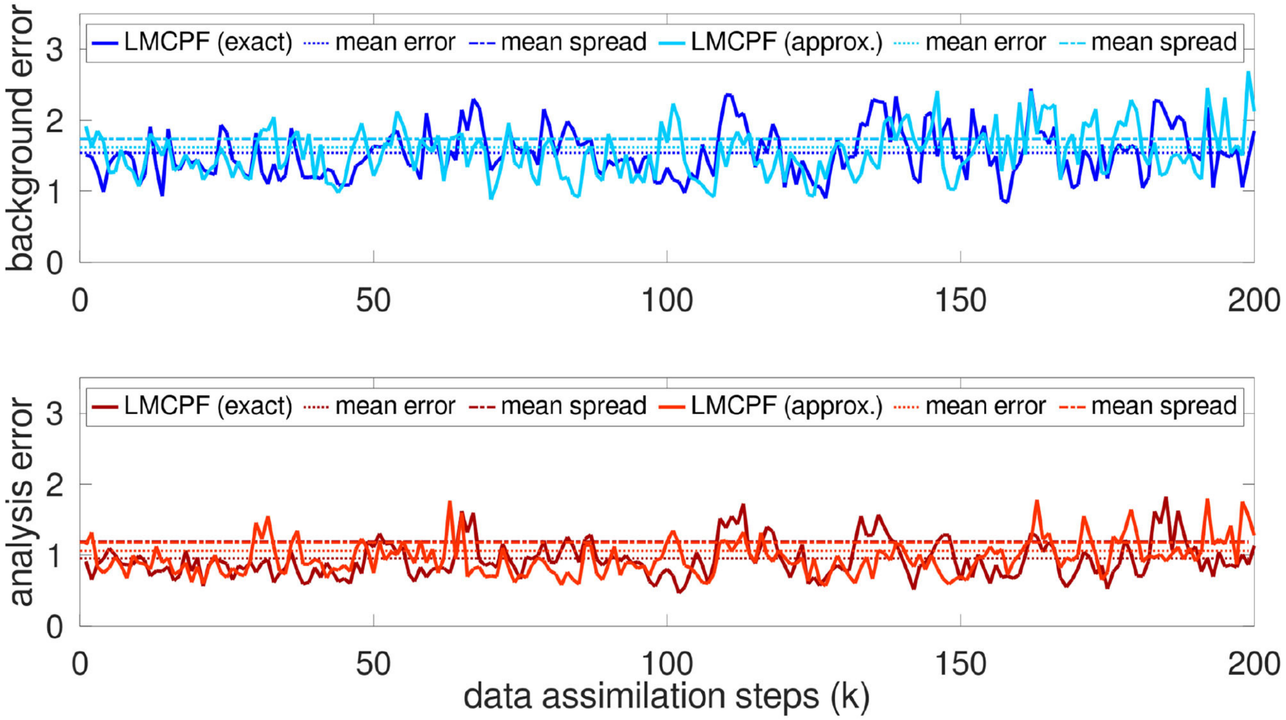


FIGURE 9 | The evolution of the background and analysis errors [see Equations (115) and (116)] for LMCPF with exact and approximate weights is illustrated for the last 200 data assimilation steps of an experiment over 1, 000 steps. For both methods, the ensemble size is set to $L = 20$. Every second variable of the 40-dimensional Lorenz 1996 model is observed. The forcing terms are set to $F^{true} = 8$ and $F = 9.5$ and the forecast length is set to $\Delta_t = 0.3$. The observation standard deviation is chosen as $\sigma_{\text{obs}} = 0.5$ and the observation error covariance matrix as diagonal matrix $R = \sigma^2_{\text{obs}} \cdot I_m$. The particle uncertainty parameter is set to $\kappa = 1.1$ for the LMCPF with exact weights and to $\kappa = 1.0$ for the LMCPF with approximate weights. The background error mean of the last 900 data assimilation steps of the LMCPF with exact weights equals $e^{(b)} \approx 1.54$ and the analysis error mean is approximately $e^{(a)} \approx 0.95$. The respective error means for the LMCPF with approximate weights are given by $e^{(b)} \approx 1.62$ and $e^{(a)} \approx 1.06$.

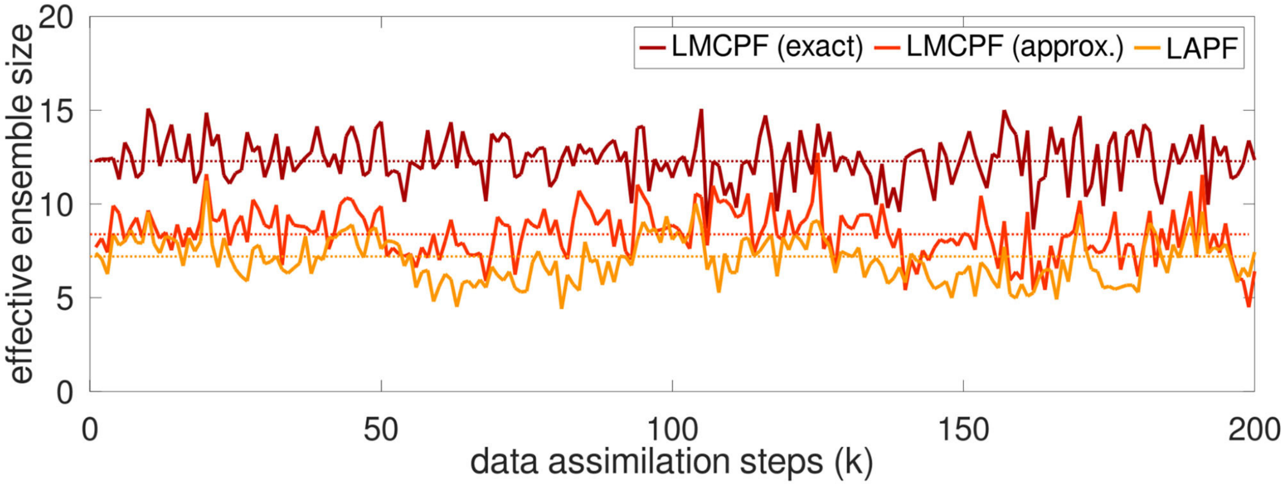


FIGURE 10 | The effective ensemble size $\bar{L}_{\text{eff}}$ defined in Equation (120) of the LMCPF method with exact and approximate weights as well as the LAPF method is shown for the last 200 steps of the data assimilation experiment described in **Figure 9**. The ensemble size is set to $L = 20$ which is the highest value $\bar{L}_{\text{eff}}$ can take on. The dotted lines denote the mean effective ensemble size over the whole experiment except a spin-up phase (last 900 data assimilation steps).

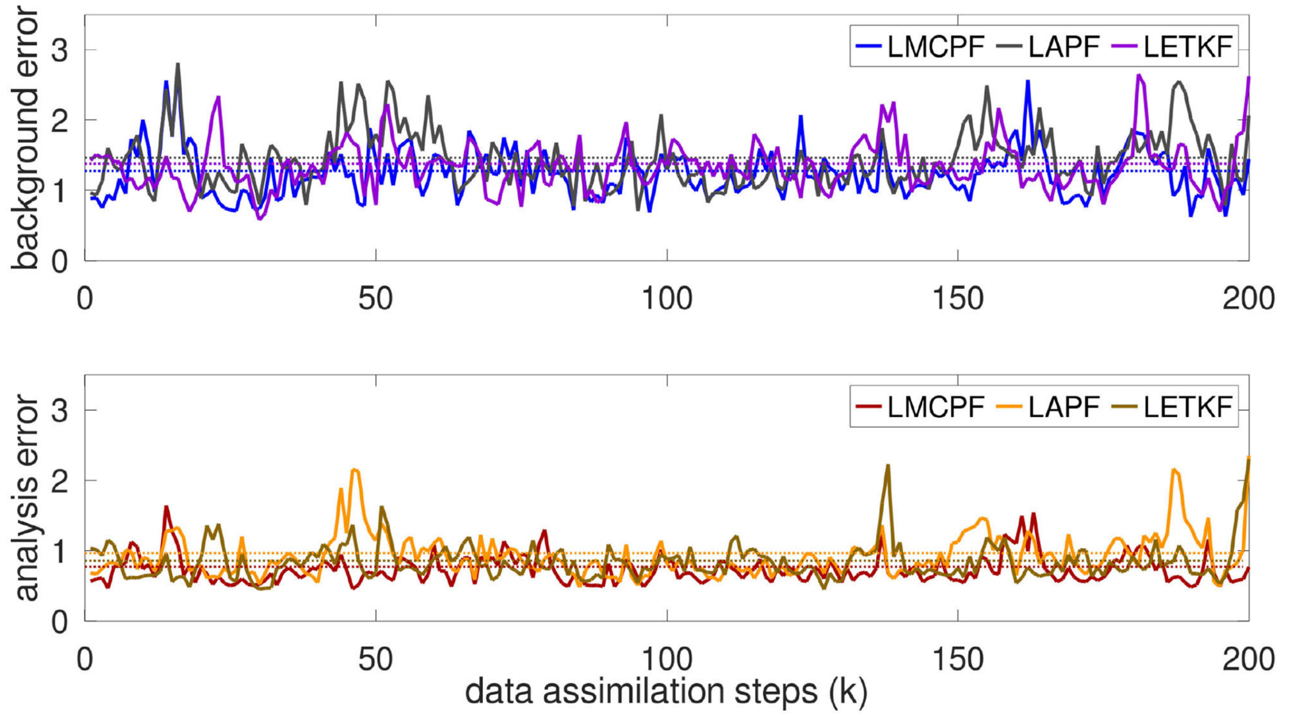


FIGURE 11 | The evolution of the background errors and analysis errors [see Equations (115) and (116)] for LMCPF ($\kappa = 1.1$), LETKF and LAPF is illustrated for the last 200 steps of an experiment over 1,000 steps. The dotted lines denote the mean errors over the whole experiment except a spin-up phase (last 900 data assimilation steps). For all methods, the ensemble size is set to $L = 20$. Every second variable of the 40-dimensional Lorenz 1996 model is observed. The forcing terms are set to $F^{\text{true}} = 8$ and $F = 9$. The forecast length is set to $\Delta_t = 0.3$. The observation standard deviation is chosen as $\sigma_{\text{obs}} = 0.5$ and the observation error covariance matrix as diagonal matrix $R = \sigma^2_{\text{obs}} \cdot I_m$. The background error mean of the last 900 data assimilation steps of the LMCPF equals $e^{(b)} \approx 1.28$ and the analysis error mean is approximately $e^{(a)} \approx 0.77$. The respective error means for the LETKF are given by $e^{(b)} \approx 1.38$ and $e^{(a)} \approx 0.86$, respectively by $e^{(b)} \approx 1.46$ and $e^{(a)} \approx 0.97$ for the LAPF.

mean and the true trajectory is shown for the LMCPF method. For the experiment, every second variable of the 40 nodes of the Lorenz 1996 model is observed. The vertical structure in **Figure 12B** indicates a lower distance of analysis mean and truth for observed variables. **Figure 12A** shows that the background errors for observed and unobserved variables are largely mixed and the vertical structure can only be guessed at some points. This results from the relatively long integration time and the large model error induced by the different model parameter $F = 9$ in the time integration of the ensemble.

In this study, we focused on the Lorenz 1996 model with 40 variables. This setting is widely used for tests of data assimilation methods and tuning of filter parameters is possible in a reasonable amount of time. Nevertheless, it is important to investigate if the particle filter methods still work for much higher dimensions. To this end, we made first experiments with respect to the Lorenz 1996 model with 1,000 variables. LAPF and LMCPF (as well as LETKF) run stably with initial ensemble size $L = 40$ and no filter divergence occured. Moreover, LAPF and LMCPF with approximate weights were already tested with respect to the global ICON model in the data assimilation framework at DWD.

6. CONCLUSION

Standard algorithms for data assimilation in the application of NWP in high-dimensional spaces are in general ensemble methods, where the ensemble describes the sample of an underlying distribution. The ensemble Kalman filter is an example for a standard algorithm, which is based on normality assumptions. However, the application of nonlinear models to a Gaussian distribution leads to a loss of the normality property in general. In future, the dynamical models used in NWP will get even more nonlinear due to higher resolution and more complex physical schemes, so that this approach might be not optimal in highly nonlinear situations. Hence, there is a need for fully nonlinear data assimilation methods, which are applicable in high dimensional spaces.

This work covers two nonlinear particle filter methods, which are already implemented and tested in the operational data assimilation system of the German Weather Service (DWD). Previous studies of the localized adaptive particle filter (LAPF; [31]) and the localized mixture coefficients particle filter (LMCPF; [32]) showed mixed results for the global NWP system at DWD. The particle filter methods were compared to the local ensemble transform Kalman filter (LETKF). With this manuscript, we examine the question if the LMCPF is able to

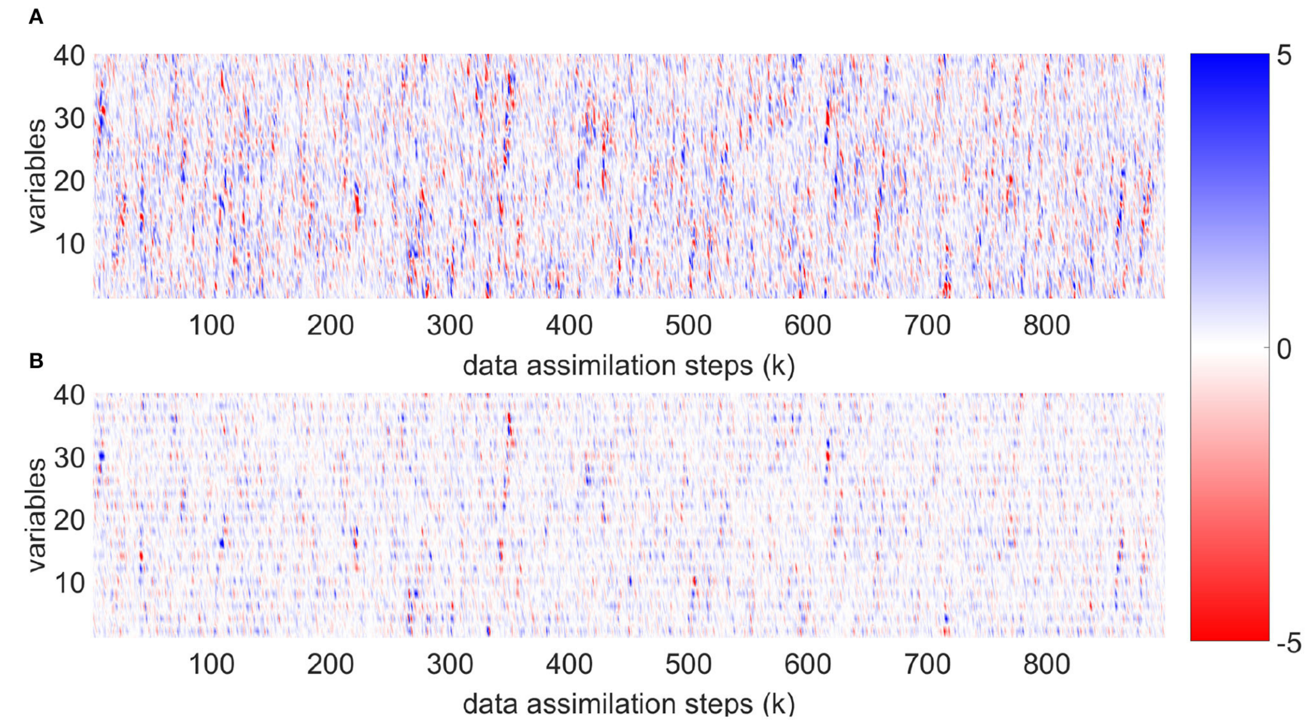


FIGURE 12 | (A) shows the difference of background mean and truth for the LMCPF method for all 40 variables for the experiment described in **Figure 11**. In **(B)**, the analysis mean minus truth is illustrated for the LMCPF method.

outperform the LETKF, with respect to a standard NWP setup and standard NWP scores for the dynamical models Lorenz 1963 and Lorenz 1996. The experiments are performed with a revised version of the LMCPF method. The exact particle weights are derived in this work. Previously, the weights were approximated by those of the LAPF. Recently, the revised method is also presented in Kotsuki et al. [39] and tested for an intermediate AGCM. The effective ensemble size is increased for the exact weights, which results in a more stable filter with respect to the parameters of the LMCPF. In case of higher effective ensemble sizes, more background information is contained, while the filter degenerates if the effective ensemble size tends to one. In this study, we demonstrated that the LMCPF is able to outperform the LETKF method with respect to the root-mean-square-error (RMSE) of background/analysis ensemble mean minus truth in case of model error for both systems. That means, the inital question, if the LMCPF is capable to outperform the LETKF within an experimental design reflecting a standard NWP setup and standard NWP scores, can be answered with yes. The experiments with Lorenz 1963 show that the longer the forecast length is chosen, which results in a higher nonlinearity, the better are the scores of LMCPF compared to LETKF. In that case, the LMCPF outperforms the LETKF for a wide range of parameter settings of the LMCPF. Even if the particle uncertainty parameter κ, which affects the ensemble spread as well as the shift toward the observation, is not perfectly adjusted, the RMSE of background ensemble mean minus truth is lower than the error of LETKF. A similar effect is visible for larger systematic model error, which is exemplarily shown with respect to the dynamical system Lorenz 1996. Moreover, further experiments for all of these localized methods, LMCPF (with exact and approximate particle weights), LAPF and LETKF, suggest, that the revised LMCPF is an improvement compared to the previous version of the LMCPF as well as the LAPF and is able to outperform the LETKF.

In the application of data assimilation methods in complex NWP systems, the behavior of the methods is overlaid by a multitude of other processes. In this work, we present the individual ingredients of the LMCPF method in one assimilation step with respect to the Lorenz 1963 model. In case of a bimodal background distribution, the analysis ensemble of the LMCPF method builds a more realistic uncertainty estimation than for the LETKF. Furthermore, the improvement of LMCPF over LAPF is demonstrated in the case of a large distance between the particles and the observation, respectively, true state. In contrast to the LAPF, the analysis ensemble, generated by the LMCPF method, is pulled stronger toward the observation due to the additional shift.

All in all, the results suggest that particle filter methods and the LMCPF in particular represent a serious alternative to the LETKF in nonlinear environments in the future. As next steps, we want to test the improved LMCPF method with respect to the global ICON model as well as the convective-scale ICON-LAM. Additionally, the application within a higher dimensional

Lorenz 1996 model (starting from $1,000$ variables) is interesting to investigate further. Moreover, we plan to focus on further scores to compare LMCPF to LETKF.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

NS, RP, and AR conceived the study. Execution of the numerical calculations were performed by NS. Writing the publication was done by NS. Revising the manuscript was done by RP and NS. All authors contributed to the article and approved the submitted version.

FUNDING

Funding is provided by Deutscher Wetterdienst (German Meteorological Service).

ACKNOWLEDGMENTS

The authors would like to thank Bastian von Harrach from Goethe-University for scientific support.

REFERENCES

1. Evensen G. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J Geophys Res Oceans*. (1994) 99:10143–62. doi: 10.1029/94JC00572
2. Evensen G, van Leeuwen PJ. An ensemble kalman smoother for nonlinear dynamics. *Mon Weather Rev*. (2000) 128:1852–67. doi: 10.1175/1520-0493(2000)128<1852:AEKSFN>2.0.CO;2
3. Evensen G. *Data Assimilation: The Ensemble Kalman Filter. Earth and Environmental Science*. 2nd ed. Dordrecht: Springer (2009). Available online at: http://books.google.de/books?id=2_zaTb_O1AkC
4. Hunt BR, Kostelich EJ, Szunyogh I. Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter. *Physica D*. (2007) 230:112–26. doi: 10.1016/j.physd.2006.11.008
5. van Leeuwen PJ. Particle filtering in geophysical systems. *Mon Weather Rev*. (2009) 137:4089–114. doi: 10.1175/2009MWR2835.1
6. Snyder C, Bengtsson T, Bickel P, Anderson J. Obstacles to high-dimensional particle filtering. *Mon Weather Rev*. (2008) 136:4629–40. doi: 10.1175/2008MWR2529.1
7. Bickel P, Li B, Bengtsson T. Sharp failure rates for the bootstrap particle filter in high dimensions. In: *Pushing the Limits of Contemporary Statistics: Contributions in Honor of Jayanta K Ghosh*. Beachwood, OH (2008). p. 318–29.
8. Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc F Radar Signal Process*. (1993) 140:107–13. doi: 10.1049/ip-f-2.1993.0015
9. van Leeuwen PJ, Künsch HR, Nerger L, Potthast R, Reich S. Particle filters for high-dimensional geoscience applications: a review. *Q J R Meteorol Soc*. (2019) 145:2335–65. doi: 10.1002/qj.3551
10. Stordal A, Karlsen H, Nævdal G, Skaug H, Vallès B. Bridging the ensemble Kalman filter and particle filters: the adaptive Gaussian mixture filter. *Comput Geosci*. (2011) 15:293–305. doi: 10.1007/s10596-010-9207-1
11. Frei M, Künsch HR. Bridging the ensemble Kalman and particle filters. *Biometrika*. (2013) 100:781–800. doi: 10.1093/biomet/ast020
12. Robert S, Künsch H. Localizing the Ensemble Kalman particle filter. *Tellus A*. (2017) 69:1–14. doi: 10.1080/16000870.2017.1282016
13. Robert S, Leuenberger D, Künsch HR. A local ensemble transform Kalman particle filter for convective-scale data assimilation. *Q J R Meteorol Soc*. (2018) 144:1279–96. doi: 10.1002/qj.3116
14. Nakano S, Ueno G, Higuchi T. Merging particle filter for sequential data assimilation. *Nonlinear Process Geophys*. (2007) 14:395–408. doi: 10.5194/npg-14-395-2007
15. Xiong X, Navon IM, Uzunoglu B. A note on the particle filter with posterior gaussian resampling. *Tellus A*. (2006) 58:456–60. doi: 10.1111/j.1600-0870.2006.00185.x
16. Tödter J, Ahrens B. A second-order exact ensemble square root filter for nonlinear data assimilation. *Mon Weather Rev*. (2015) 143:1347–67. doi: 10.1175/MWR-D-14-00108.1
17. Bishop CH, Etherton BJ, Majumdar SJ. Adaptive sampling with the ensemble transform Kalman filter. Part I: theoretical aspects. *Mon Weather Rev*. (2001) 129:420–36. doi: 10.1175/1520-0493(2001)129<0420:ASWTET>2.0.CO;2
18. Reich S. A nonparametric ensemble transform method for bayesian inference. *SIAM J Scientific Comput*. (2013) 35:A2013–24. doi: 10.1137/130907367
19. Neal RM. Sampling from multimodal distributions using tempered transitions. *Stat Comput*. (1996) 6:353–66. doi: 10.1007/BF00143556
20. Del Moral P, Doucet A, Jasra A. Sequential monte carlo samplers. *J R Stat Soc B*. (2006) 68:411–36. doi: 10.1111/j.1467-9868.2006.00553.x
21. Emerick AA, Reynolds AC. Ensemble smoother with multiple data assimilation. *Comput Geosci*. (2013) 55:3–15. doi: 10.1016/j.cageo.2012.03.011
22. Beskos A, Crisan D, Jasra A. On the stability of sequential Monte Carlo methods in high dimensions. *Ann Appl Probab*. (2014) 24:1396–445. doi: 10.1214/13-AAP951
23. van Leeuwen PJ. Nonlinear ensemble data assimilation for the ocean. In: *Seminar on Recent Developments in Data Assimilation for Atmosphere Ocean, 8–12 September 2003 ECMWF*. Shinfield Park; Reading: ECMWF (2003). p. 265–86.
24. Reich S. A Gaussian-mixture ensemble transform filter. *Q J R Meteorol Soc*. (2012) 138:222–33. doi: 10.1002/qj.898
25. Reich S, Cotter C. *Probabilistic Forecasting and Bayesian Data Assimilation*. Cambridge: Cambridge University Press (2015).
26. Liu Q, Wang D. Stein variational gradient descent: a general purpose Bayesian inference algorithm. In: Lee DD, Sugiyama M, Luxburg UV, Guyon I, Garnett R, editors. *Advances in Neural Information Processing Systems, Vol. 29*. Curran Associates, Inc. (2016). p. 2378–86. Available online at: https://proceedings.neurips.cc/paper/2016/file/b3ba8f1bee1238a2f37603d90b58898d-Paper.pdf (accessed June 10, 2022).
27. Lu J, Lu Y, Nolen J. Scaling limit of the stein variational gradient descent: the mean field regime. *SIAM J Math Anal*. (2019) 51:648–71. doi: 10.1137/18M1187611
28. Poterjoy J. A localized particle filter for high-dimensional nonlinear systems. *Mon Weather Rev*. (2016) 144:59–76. doi: 10.1175/MWR-D-15-0163.1
29. Poterjoy J, Sobash RA, Anderson JL. Convective-Scale data assimilation for the weather research and forecasting model using the local particle filter. *Mon Weather Rev*. (2017) 145:1897–918. doi: 10.1175/MWR-D-16-0298.1
30. Penny SG, Miyoshi T. A local particle filter for high-dimensional geophysical systems. *Nonlinear Process Geophys*. (2016) 23:391–405. doi: 10.5194/npg-23-391-2016
31. Potthast R, Walter A, Rhodin A. A localized adaptive particle filter within an operational nwp framework. *Mon Weather Rev*. (2019) 147:345–62. doi: 10.1175/MWR-D-18-0028.1
32. Rojahn, A., Schenk, N., van Leeuwen, P. J., and Potthast, R. (2022). *Particle filtering and Gaussian mixtures – On a localized mixture coefficients particle filter (LMCPF) for global NWP*. Preprint. doi: 10.48550/arXiv.2206.07433
33. Alspach D, Sorenson H. Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Trans Automat Contr*. (1972) 17:439–48. doi: 10.1109/TAC.1972.1100034

34. Anderson JL, Anderson SL. A monte carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon Weather Rev*. (1999) 127:2741–58. doi: 10.1175/1520-0493(1999)127<2741:AMCIOT>2.0.CO;2
35. Chen R, Liu J. Mixture Kalman Filter. *J Roy Statist Soc Ser B*. (2000) 62:493–508. doi: 10.1111/1467-9868.00246
36. Bengtsson T, Snyder C, Nychka D. Toward a nonlinear ensemble filter for high-dimensional systems. *J Geophys Res Atmospheres*. (2003) 108:8775. doi: 10.1029/2002JD002900
37. Kotecha JH, Djuric PM. Gaussian particle filtering. *IEEE Trans Signal Process*. (2003) 51:2592–601. doi: 10.1109/TSP.2003.816758
38. Hoteit I, Pham DT, Triantafyllou G, Korres G. A new approximate solution of the optimal nonlinear filter for data assimilation in meteorology and oceanography. *Mon Weather Rev*. (2008) 136:317–34. doi: 10.1175/2007MWR1927.1
39. Kotsuki S, Miyoshi T, Kondo K, Potthast R. *A Local Particle Filter and Its Gaussian Mixture Extension: Comparison With the LETKF Using an Intermediate AGCM*. (2022). doi: 10.5194/gmd-2022-69
40. Zängl G, Reinert D, Rípodas P, Baldauf M. The ICON (ICOsahedral Non-hydrostatic) modelling framework of DWD and MPI-M: description of the non-hydrostatic dynamical core. *Q J R Meteorol Soc*. (2015) 141:563–79. doi: 10.1002/qj.2378
41. Lorenz EN. Deterministic nonperiodic flow. *J Atmosphere Sci*. (1963) 20:130–41. doi: 10.1175/1520-04691963020<0130:DNF>gt;2.0.CO;2
42. Goodliff M, Amezcua J, Leeuwen PJV. Comparing hybrid data assimilation methods on the Lorenz 1963 model with increasing non-linearity. *Tellus A*. (2015) 67:26928. doi: 10.3402/tellusa.v67.26928
43. Miller RN, Ghil M, Gauthiez F. Advanced data assimilation in strongly nonlinear dynamical systems. *J Atmospher Sci*. (1994) 51:1037–56. doi: 10.1175/1520-0469(1994)051<1037:ADAISN>2.0.CO;2
44. Lorenz EN. Predictability: a problem partly solved. In: *Seminar on Predictability, 4-8 September 1995, vol. 1. ECMWF*. Shinfield Park, Reading: ECMWF (1995). p. 1–18.
45. Li H, Kalnay E, Miyoshi T. Simultaneous estimation of covariance inflation and observation errors within an ensemble Kalman filter. *Q J R Meteorol Soc*. (2009) 135:523–33. doi: 10.1002/qj.371
46. van Leeuwen PJ. Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Q J R Meteorol Soc*. (2010) 136:1991–9. doi: 10.1002/qj.699
47. Frei M, Künsch HR. Sequential state and observation noise covariance estimation using combined ensemble Kalman and particle filters. *Mon Weather Rev*. (2012) 140:1476–95. doi: 10.1175/MWR-D-10-05088.1
48. Nerger L, Janjić T, Schröter J, Hiller W. A regulated localization scheme for ensemble-based Kalman filters. *Q J R Meteorol Soc*. (2012) 138:802–12. doi: 10.1002/qj.945
49. Papoulis A, Pillai SU. *Probability, Random Variables, and Stochastic Processes. 3rd Edn*. Boston, MA: McGraw-Hill (1991).
50. Kirchgessner P, Nerger L, Bunse-Gerstner A. On the choice of an optimal localization radius in ensemble Kalman filter methods. *Mon Weather Rev*. (2014) 142:2165–75. doi: 10.1175/MWR-D-13-00246.1
51. Gaspari G, Cohn SE. Construction of correlation functions in two and three dimensions. *Q J R Meteorol Soc*. (1999) 125:723–57. doi: 10.1002/qj.49712555417
52. Nakamura G, Potthast R. *Inverse Modeling: An Introduction to the Theory and Methods of Inverse Problems and Data Assimilation*. Bristol: IOP Publishing (2015).
53. Carpenter J, Cliffordy P, Fearnhead P. An improved particle filter for non-linear problems. *IEE Proc Radar Sonar Navig*. (2000) 146:2–7. doi: 10.1049/ip-rsn:19990255
54. Anderson BDO, Moore JB. *Optimal Filtering*. Hoboken, NJ: Prentice-Hall (1979).
55. Desroziers G, Berre L, Chapnik B, Poli P. Diagnosis of observation, background and analysis-error statistics in observation space. *Q J R Meteorol Soc*. (2005) 131:3385–96. doi: 10.1256/qj.05.108
56. Feng J, Wang X, Poterjoy J. A comparison of two local moment-matching nonlinear filters: local particle filter (LPF) and local nonlinear ensemble transform filter (LNETF). *Mon Weather Rev*. (2020) 148:4377–95. doi: 10.1175/MWR-D-19-0368.1
57. Poterjoy J, Wicker L, Buehner M. Progress toward the application of a localized particle filter for numerical weather prediction. *Mon Weather Rev*. (2019) 147:1107–26. doi: 10.1175/MWR-D-17-0344.1
58. Schraff C, Reich H, Rhodin A, Schomburg A, Stephan K, Periáñez A, et al. Kilometre-scale ensemble data assimilation for the COSMO model (KENDA). *Q J R Meteorol Soc*. (2016) 142:1453–472. doi: 10.1002/qj.2748
59. Evensen G. The ensemble kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*. (2003) 53:343–67. doi: 10.1007/s10236-003-0036-9
60. Lei J, Bickel P. A moment matching ensemble filter for nonlinear non-gaussian data assimilation. *Mon Weather Rev*. (2011) 139:3964–73. doi: 10.1175/2011MWR3553.1

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

TYPE Original Research
PUBLISHED 16 September 2022
DOI 10.3389/fams.2022.892098

Mathematical analysis of cholera typhoid co-infection transmission dynamics

Lunga Matsebula* and Farai Nyabadza

Department of Mathematics and Applied Mathematics, University of Johannesburg, Johannesburg, South Africa

Check for updates

OPEN ACCESS

EDITED BY
Pierre Magal,
Université de Bordeaux, France

REVIEWED BY
Feng-Bin Wang,
Chang Gung University, Taiwan
Gui-Quan Sun,
North University of China, China

*CORRESPONDENCE
Lunga Matsebula
lmatsebula@uj.ac.za

SPECIALTY SECTION
This article was submitted to Mathematical Biology, a section of the journal Frontiers in Applied Mathematics and Statistics

RECEIVED 08 March 2022
ACCEPTED 29 August 2022
PUBLISHED 16 September 2022

CITATION
Matsebula L and Nyabadza F (2022) Mathematical analysis of cholera typhoid co-infection transmission dynamics. *Front. Appl. Math. Stat.* 8:892098. doi: 10.3389/fams.2022.892098

Typhoid fever and cholera remain a huge public health problem on the African continent due to deteriorating infrastructure and declining funding for infrastructure development. The diseases are both caused by bacteria, and they are associated with poor hygiene and waste disposal systems. In this paper, we consider a nonlinear system of ordinary differential equations for the co-infection of typhoid and cholera in a homogeneously mixing population. The model's steady states are determined and analyzed in terms of the model's reproduction number. Impact analysis—how the diseases impact on each other—is carried out. Numerical simulations and sensitivity analysis are also given. The results show that the control of the diseases should be carried out in tandem for the greatest impact of disease control. The results have important implications in the management of the two diseases.

KEYWORDS
cholera, typhoid, co-infection, stability, basic reproduction number, impact analysis

1. Introduction

Cholera, an acute gastro-intestinal water-borne infection, is caused by the bacterium *Vibrio Cholerae*, V. *cholerae* O1 or O139. Some of the symptoms are vomiting and diarrhea. If treatment is delayed, it can lead to severe dehydration and death within a few hours. The disease has two modes of transmission: direct and indirect transmission. Direct transmission (human-human) is very uncommon, whilst indirect transmission (environment-human), which occurs through the ingestion of contaminated food or water [1], is more frequent. Known estimates of the incubation period for the cholera disease is 1.4 days [2]. On the other hand, the *Salmonella Typhi* bacteria is responsible for causing the life threatening typhoid fever disease. Cholera and typhoid fever have the same transmission modes. The recticuloendothelical system, the intestinal lymphoid, and the gall bladder are severely damaged by the typhoid fever disease. Once a susceptible individuals is infected by the disease, roughly 19 days are required for the disease to incubate within the host [3].

Mathematical models have been used for the past decades to give insights into the transmission dynamics of co-infections within the human population. Akinyi et al. [4], showed that whenever the basic reproduction number is lowered to below one, then the malaria and the pneumonia cases will be reduced in a model of malaria-pneumonia coinfection. Onyinge et al. [5] modeled the co-dynamics of pneumonia and HIV, and they showed that the model was mathematically and epidemiologically sound; Mushayabasa et al. [6] modeled malaria-typhoid co-infection and demonstrated that a typhoid outbreak will inevitably lead to a spike in the malaria cases.

A number of mathematical models on typhoid have been proposed by a number of researchers. Mushayabasa [7], modeled how vaccines can help mitigate the spread of typhoid in Ghana. Pitzer et al. [8], extended the work in Mushayabasa [7] by applying the model to South Asia. Khan et al. [9], studied the typhoid disease with a saturated incidence rate.

To the best of our knowledge, the co-dynamics of typhoid and cholera have not been investigated in the literature. A recent outbreak of these two infections in Zimbabwe prompted this theoretical inquiry into how these infections interact. Due to the complicated nature of the co-infection model, we begin our analysis by studying the underlying sub-models; namely, the cholera only and the typhoid only sub-models. For each of the models, a number of pertinent questions are investigated. The questions explored include: Which factors in the models are key to decreasing the prevalence of each disease and the co-infection? Within the population, are the infections in competition with each other, or are they symbiotic? The implications of the results to the public health are discussed.

The paper is arranged as follows; the development of the model and the properties of the basic reproduction number are established in Section 2. Section 3 contains the stability analysis of the model at the fixed points. Numerical simulations and parameter estimations are done in section 4. Section 5 concludes the articles.

2. Methodology

2.1. Model development

Our typhoid cholera co-infection model partitions the human population $N(t)$, at time t, into a susceptible class $S(t)$, a cholera infection class $I_c(t)$, a typhoid infection class $I_t(t)$, a coinfection class $I_{ct}(t)$, a cholera recovery class $R_c(t)$, a typhoid recovery class $R_c(t)$, and a coinfection recovery class $R_{ct}(t)$. Thus,

$$N(t) = S(t) + I_c(t) + R_c(t) + I_t(t) + R_t(t) + I_{ct}(t) + R_{ct}(t).$$

The bacterial concentration of *Salmonella Typhi*, $B_t(t)$, and *Vibrio Cholerae*, $B_c(t)$, in the environment are incorporated into the model as well. The formulation of this model is an extension to the work carried out by Matsebula et al. [10].

Since the incubation periods of the two infections are different, we assume that dually infected individuals can only transmit either cholera or typhoid but not both infections simultaneously. Transmission of cholera to susceptible individuals occurs in one of two routes—the direct transmission route (human-to-human) and the indirect transmission route (envirnment-to-human). The rates of the transmission routes, respectively, are given by

$$\lambda_{c_1} = \frac{\beta_{c_1}(I_c + \eta_c I_{ct})}{N}, \qquad \lambda_{c_2} = \frac{\beta_{c_2} B_c}{B_c + \kappa_c}.$$

The parameter β_{c_1} denotes the person-to-person cholera transmission. The effective contact rate for cholera multiplied by the probability of cholera transmission per contact gives the person to person cholera transmission. The modification parameter η_c, accounts for the relative infectiousness of individuals in class I_c relative to individuals in class I_{ct}. We assume that $\eta_c \in (0, 1)$. This assumption is motivated by the fewer numbers of co-infected individuals as compared to those infected with cholera only. The parameter β_{c_2} denotes the environment-to-humans per capita contact rate and the *Vibrio Cholerae* in the contaminated environment, whilst the parameter κ_c denotes the half saturation constant of the *Vibrio Cholerae*. The *half saturation constant* is the bacterial concentration that is required to support half of the maximum rate, β_{c_2}.

Similarly, the transmission of typhoid to susceptible individuals occurs in one of two routes—the direct transmission route (human-to-human) and the indirect transmission route (envirnment-to-human). The rates of the transmission routes, respectively, are given by

$$\lambda_{t_1} = \frac{\beta_{t_1}(I_t + \eta_t I_{ct})}{N}, \qquad \lambda_{t_2} = \frac{\beta_{t_2} B_t}{B_t + \kappa_t}.$$

The parameter β_{t_1} denotes the effective person-to-person typhoid transmission rate. The effective contact rate for typhoid multiplied by the probability of typhoid transmission per contact gives the person to person typhoid transmission. The modification parameter η_t, accounts for the relative infectiousness of individuals in class I_t relative to individuals in class I_{ct}. We also assume that $\eta_t \in (0, 1)$ following the assumptions given in the cholera infection dynamics. The per capita contact rate between the susceptibles and *Salmonela typhi* is represented by β_{t_2}, and the half saturation constant for λ_{c_2} is κ_t.

Transmission of cholera to typhoid infected individuals occurs in one of two routes—the direct transmission route (human-to-human) and the indirect transmission route (environment-to-human). The rates of the transmission routes, respectively, are given by

$$\lambda_{c_3} = \frac{\beta_{c_3}(I_c + \eta_c I_{ct})}{N}, \qquad \lambda_{c_4} = \frac{\beta_{c_4} B_c}{B_c + \kappa_c}.$$

The parameter β_{c_3} denotes the effective person-to-person cholera transmission rate of individuals in class I_t. The parameter β_{c_4} denotes the environment-to-humans per capita contact rate for individuals in class I_t and the *Vibrio Cholerae* in the contaminated environment. Transmission of typhoid to

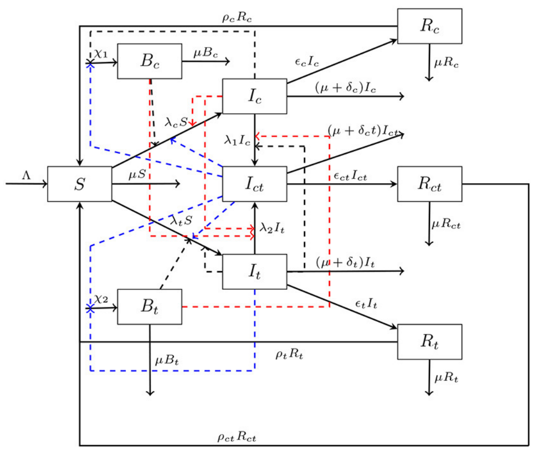


FIGURE 1
The cholera typhoid co-infection compartmental model. For the concise presentation of our model flow diagram, we make use of the following expressions: $\chi_1 = g_c B_c\left(1-\frac{B_c}{k_c}\right)+\alpha_c I_c+\theta_c I_{ct}$, $\chi_2 = g_t B_t\left(1-\frac{B_t}{k_t}\right)+\alpha_t I_t+\theta_t I_{ct}$, $\lambda_c=\lambda_{c_1}+\lambda_{c_2}$, $\lambda_t=\lambda_{t_1}+\lambda_{t_2}$, $\lambda_1=\lambda_{t_3}+\lambda_{t_4}$, $\lambda_2=\lambda_{c_3}+\lambda_{c_4}$.

cholera infected individuals occurs in one of two routes—the direct transmission route (human-to-human) and the indirect transmission route (environment-to-human). The rates of the transmission routes, respectively, are given by

$$\lambda_{t_3} = \frac{\beta_{t_3}(I_t+\eta_t I_{ct})}{N}, \qquad \lambda_{t_4} = \frac{\beta_{t_4} B_t}{B_t+\kappa_t}.$$

The parameter β_{t_3} denotes the person-to-person typhoid transmission rate of individuals in class I_c. The parameter β_{t_4} denotes the environment-to-humans per capita contact rate for individuals in class I_c and the *Salmonella Typhi* in the contaminated environment.

Infected individuals in classes I_c, I_t and I_{ct} experience disease related death at rates given, respectively by δ_c, δ_t and δ_{ct}. Individuals in the infectious states I_c and I_t, respectively, excrete *Vibrio Cholerae* bacteria and *Salmonella Typhi* bacteria into the environment at rates α_c and α_t. Coinfected individuals shed *Vibrio Cholerae* and *Salmonella Typhi* into the environments at rates θ_c and θ_t, respectively. Infection is assume to confer temporary immunity. The cholera and typhoid immunity wanes at rates ρ_c, ρ_t and ρ_{ct}.

The generation rate of *Vibrio Cholerae* is $g_c B_c\left(1-\frac{B_c}{k_c}\right)$, and its growth is enhanced by cholera infected individuals and the coinfected individuals that are shedding into the environment. The generation rate of *Salmonella Typhi* is $g_t B_t\left(1-\frac{B_t}{k_t}\right)$ and its growth is enhanced by typhoid infected individuals and the coinfected individuals that are shedding into the environment. We assume that the *Vibrio Cholerae* and the *Salmonella Typhi* bacteria in the environment are respectively removed by interventions such as improved sanitation and treatment of contaminated environments at rates μ_c and μ_t. The parameter Λ represents the recruitment into the susceptibles, while the parameter μ represents the natural death rate. It is assumes that individuals mix homogeneously and that they are indistinguishable in each of the classes. The model diagram is shown in Figure 1.

The dynamical system associated with the schematic diagram in Figure 1 is;

$$\begin{aligned}
\frac{dS}{dt} &=\Lambda-(\lambda_{c_1}+\lambda_{c_2}+\lambda_{t_1}+\lambda_{t_2})S-\mu S+\rho_c R_c+\rho_t R_t\\
&\quad+\rho_{ct}R_{ct},\\
\frac{dI_c}{dt} &=(\lambda_{c_1}+\lambda_{c_2})S-(\lambda_{t_3}+\lambda_{t_4})I_c-(\mu+\delta_c+\epsilon_c)I_c,\\
\frac{dI_t}{dt} &=(\lambda_{t_1}+\lambda_{t_2})S-(\lambda_{c_3}+\lambda_{c_4})I_t-(\mu+\delta_t+\epsilon_t)I_t,\\
\frac{dI_{ct}}{dt} &=(\lambda_{t_3}+\lambda_{t_4})I_c+(\lambda_{c_3}+\lambda_{c_4})I_t-(\mu+\delta_{ct}+\epsilon_{ct})I_{ct},\\
\frac{dR_c}{dt} &=\epsilon_c I_c-(\mu+\rho_c)R_c, \qquad (1)\\
\frac{dR_t}{dt} &=\epsilon_t I_t-(\mu+\rho_t)R_t,\\
\frac{dR_{ct}}{dt} &=\epsilon_{ct} I_{ct}-(\mu+\rho_{ct})R_{ct},\\
\frac{dB_c}{dt} &=g_c B_c\left(1-\frac{B_c}{k_c}\right)+\alpha_c I_c+\theta_c I_{ct}-\mu_c B_c,\\
\frac{dB_t}{dt} &=g_t B_t\left(1-\frac{B_t}{k_t}\right)+\alpha_t I_t+\theta_t I_{ct}-\mu_t B_t,
\end{aligned}$$

with initial conditions

$$\begin{aligned}
S(0)&=S_0>0, \quad B_c(0)=B_{c0}\geq 0, \quad B_t(0)=B_{t0}\geq 0,\\
I_c(0)&=I_{c0}\geq 0, I_t(0)=I_{t0}\geq 0, \quad I_{ct}(0)=I_{ct0}\geq 0,\\
R_c(0)&=R_{c0}\geq 0, \quad R_t(0)=R_{t0}\geq 0, R_{ct}(0)=R_{ct0}\geq 0.
\end{aligned}$$

2.2. Cholera only model

We define the cholera only model as the model obtained from setting all the typhoid classes and its associated parameters to zero. We thus have the following

$$\begin{aligned}
\frac{dS}{dt} &= \Lambda-(\tilde{\lambda}_{c_1}+\lambda_{c_2})S-\mu S+\rho_c R_c,\\
\frac{dI_c}{dt} &= (\tilde{\lambda}_{c_1}+\lambda_{c_2})S-q_c I_c,\\
\frac{dR_c}{dt} &= \epsilon_c I_c-(\mu+\rho_c)R_c, \qquad (2)\\
\frac{dB_c}{dt} &= g_c B_c\left(1-\frac{B_c}{k_c}\right)+\alpha_c I_c-\mu_c B_c,
\end{aligned}$$

where

$$\tilde{\lambda}_{c_1}=\frac{\beta_{c_1} I_c}{N_c}, \quad q_c=\mu+\delta_c+\epsilon_c, \quad N_c=S+I_c+R_c,$$

with initial conditions

$$S(0) = S_0 > 0, \quad B_c(0) = B_{c0} \geq 0, I_c(0) = I_{c0} \geq 0,$$
$$R_c(0) = R_{c0} \geq 0.$$

2.2.1. Boundedness and non-negative trajectories

We argue that model (Equation 2) yields bounded-non negative-trajectories in this section provided the initial conditions are non-negative.

Theorem 1. *All solutions of the cholera only sub-model (Equation 2) are non-negative if all the initial conditions are non-negative.*

Proof. Define $t_1 = \sup\left\{t > 0 | S(\tau_1) > 0, I_c(\tau_1) \geq 0, R_c(\tau_1) \geq 0, B_c(\tau_1) \geq 0, \forall \tau_1 \in [0, t]\right\}$. It follows $t_1 > 0$ since
$S_0 > 0, I_{c0} \geq 0, R_{c0} \geq 0, B_{c0} \geq 0.$
Assume $t_1 < \infty$, then $S(t_1) > 0, I_c(t_1) = 0, R_c(t_1) = 0, B_c(t_1) = 0$. Applying variation of constants to

$$\frac{dS}{dt} = \Lambda - (\tilde{\lambda}_{c_1} + \lambda_{c_2})S - \mu S + \rho_c R_c,$$

yields

$$S(t_1) = \int_0^{t_1} f(r) \exp\left(-\int_r^{t_1} P(x)dx\right) dr$$
$$+ S_0 \exp\left(-\int_0^{t_1} P(x)dx\right),$$

where $P(x) = (\tilde{\lambda}_{c_1} + \lambda_{c_2} + \mu)$ and $f(r) = \Lambda + \rho_c R_c$. Clearly,

$$S(t_1) > 0$$

Since $f(r) > 0$ and $P(x) > 0$ when $x, r \in [0, t_1]$. Similarly, $I_c(t_1) > 0$ and $R_c(t_1) > 0$. This produces a contradiction, hence $t_1 = \infty$.

Theorem 2. *All solutions of the cholera only sub-model (Equation 2) are bounded within Ω whenever $g_c \geq \mu_c$.*

Proof. The time derivative of the population for the cholera model (Equation 2) is bounded above by

$$\frac{dN_c}{dt} = \Lambda - \mu N_c - \delta_c I_c \leq \Lambda - \mu N_c,$$

Upper bounds for the human population, $N_c(t)$, are obtained by integrating the separable differential inequality as follows,

$$N_c \leq \frac{\Lambda - M\exp(-\mu t)}{\mu} \leq \frac{\Lambda}{\mu}.$$

By extension, Λ/μ is also the upper bound for each of the human classes. Whereas, owing to $I_c \leq N_c \leq \Lambda/\mu$, an upper bound for the bacterial classes can be obtained as follows,

$$\frac{dB_c}{dt} = g_c B_c\left(1 - \frac{B_c}{k_c}\right) + \alpha_c I_c - \mu_c B_c \leq g_c B_c\left(1 - \frac{B_c}{k_c}\right)$$
$$+ \alpha_c \frac{\Lambda}{\mu} - \mu_c B_c. \tag{3}$$

From inequality (3), if

$$B_c \geq \alpha_c \frac{\Lambda}{\mu}, \tag{4}$$

Where $\alpha_c \frac{\Lambda}{\mu}$ is the maximum shedding rate from the cholera infected individuals, then

$$\frac{dB_c}{dt} \leq (g_c - \mu_c)B_c - \tfrac{g_c}{k_c}B_c^2 + B_c$$
$$= (g_c - \mu_c + 1)B_c\left(1 - \frac{g_c B_c}{k_c(g_c - \mu_c + 1)}\right). \tag{5}$$

The constant

$$\frac{k_c(g_c - \mu_c + 1)}{g_c}, \tag{6}$$

is the upper bound for the differential inequality (Equation 5) since (Equation 5) is the logistic growth model with carrying capacity (Equation 6). For some $t \geq 0$, $(\alpha_c + \theta_c)\Lambda/\mu$ is an upper bound for B_c whenever (Equation 4) is false, whilst B_c is bounded above by Equation (6) for the rest of the time points in the domain of B_c if (Equation 4) is true. Thus, in both cases, $B_c \leq \max\left\{\frac{k_c(g_c - \mu_c + 1)}{g_c}, \alpha_c \frac{\Lambda}{\mu}\right\}$.

Within the feasible region,

$$\Omega_c = \left\{(S, I_c, R_c, B_c)\middle| 0 \leq N_c \leq \frac{\Lambda}{\mu}\right],$$
$$B_c \in \left[0, \max\left\{\frac{k_c(g_c - \mu_c + 1)}{g_c}, \alpha_c \frac{\Lambda}{\mu}\right\}\right]\Bigg\},$$

We have summarized the results on the boundedness and positivity of the solutions of the cholera only sub-model 2.

2.2.2. The stability of the disease free equilibrium and the reproduction number, $\mathcal{R}_C$

The disease free equilibria of system (Equation 2) is given by

$$\mathbf{x}_0 = (S, I_c, R_c, B_c) = \left(\frac{\Lambda}{\mu}, 0, 0, 0\right),$$
$$\mathbf{x}_1 = (S, I_c, R_c, B_c) = (c_1, 0, 0, c_2),$$

where $c_1 = \dfrac{\Lambda(g_c-\mu_c)k_c + g_c\kappa_c}{(\mu+\beta_{c_2})(g_c-\mu_c)k_c + \mu\kappa_c gc}$ and $c_2 = \dfrac{g_c-\mu_c}{g_c}k_c$. The equilibrium $\mathbf{x}_1 > \mathbf{0}$ if $g_c > \mu_c$. The Jacobian of dynamical system (Equation 2) is given by

$$J(\mathbf{x}_0) = \begin{pmatrix} -\mu & -\beta_{c_1} & \rho_c & -\frac{\Lambda\beta_{c_2}}{\mu\kappa_c} \\ 0 & \beta_{c_1}-q_c & 0 & \frac{\Lambda\beta_{c_2}}{\mu\kappa_c} \\ 0 & \epsilon & -(\mu+\rho_c) & 0 \\ 0 & \alpha_c & 0 & g_c-\mu_c \end{pmatrix}.$$

The dynamical system (Equation 2) is locally asymptotically stable if all four of its eigenvalues have negative real parts. Two of the eigenvalues for the Jacobian, J, are $\lambda_1 = -\mu$ and $\lambda_2 = -(\mu+\rho_c)$. The other two eigenvalues for J are the eigenvalues from the sub-matrix

$$\bar{J} = \begin{pmatrix} \beta_{c_1}-q_c & \frac{\Lambda\beta_{c_2}}{\mu\kappa_c} \\ \alpha_c & g_c-\mu_c \end{pmatrix}.$$

The characteristic equation for matrix $\bar{J}$ is $\lambda^2 + \bar{\nu}_1\lambda + \bar{\nu}_2$, where

$$\begin{aligned} \bar{\nu}_1 &= -\left((g_c-\mu_c)+(\beta_{c_1}-q_c)\right), \\ \bar{\nu}_2 &= (\beta_{c_1}-q_c)(g_c-\mu_c)\left(1-\frac{\alpha_c\beta_{c_2}\Lambda}{(\beta_{c_1}-q_c)(g_c-\mu_c)\kappa_c\mu}\right) \\ &= (\beta_{c_1}-q_c)(g_c-\mu_c)\left(1-\mathcal{R}_C\right), \end{aligned}$$

and

$$\mathcal{R}_C = \frac{\alpha_c\beta_{c_2}\Lambda}{\kappa_c\mu\mu_c q_c(1-\mathcal{R}_h)(1-\mathcal{R}_b)}, \quad \mathcal{R}_h = \frac{\beta_{c_1}}{q_c} \quad \mathcal{R}_b = \frac{g_c}{\mu_c}.$$

The constants $\mathcal{R}_b$ and $\mathcal{R}_h$ are *the bacterial regeneration threshold* and *the human-to-human sub reproduction number*, respectively. The constant $\mathcal{R}_C$ is the so-called *basic reproduction number* for the system (Equation 2). Clearly, $\mathcal{R}_h, \mathcal{R}_b < 1$ or $\mathcal{R}_h, \mathcal{R}_b > 1$ if and only if $\mathcal{R}_C > 0$.

It follows from the Routh Hurwitz criterion that the two eigenvalues of $\bar{J}$ have negative real parts if $\bar{\nu}_1, \bar{\nu}_2 > 0$. It is easy to see that $\bar{\nu}_1, \bar{\nu}_2 > 0$ if $\mathcal{R}_b < 1, \mathcal{R}_h < 1$ and $\mathcal{R}_C < 1$. Hence, a positive basic reproduction number for system (Equation 2) that is less than unity implies that the system is locally asymptotically stable at the disease free equilibrium.

2.2.3. Endemic equilibrium

Setting the derivatives of the classes to zero gives the endemic equilibrium for the cholera only sub-model (Equation 2). Let $\lambda = \tilde{\lambda}_{c_1} + \lambda_{c_2}$.

$$0 = \Lambda - (\lambda^* + \mu)S^* + \rho_c R_c^*, \tag{7}$$

$$0 = \lambda^* S^* - q_c I_c^*, \tag{8}$$

$$0 = \epsilon_c I_c^* - (\mu+\rho_c)R_c^*, \tag{9}$$

$$0 = g_c B_c^*\left(1-\frac{B_c^*}{k_c}\right) + \alpha_c I_c^* - \mu_c B_c^*. \tag{10}$$

From Equation (8),

$$S^* = \frac{q_c}{\lambda^*}I_c^*.$$

From Equation (7),

$$R_c^* = \frac{1}{\rho_c}\left[\frac{\left(\lambda^*+\mu\right)q_c I_c^*}{\lambda^*} - \Lambda\right].$$

Consider (Equation 9),

$$\epsilon I_c^* - \frac{(\mu+\rho_c)}{\rho_c}\left[\frac{\left(\lambda^*+\mu\right)q_c I_c^*}{\lambda^*} - \Lambda\right] = 0,$$

therefore

$$I_c^* = \frac{\lambda^*\Lambda\left(\mu+\rho_c\right)}{q_c\left(\lambda^*+\mu\right)\left(\mu+\rho_c\right) - \epsilon_c\rho_c\lambda^*}.$$

Given that

$$\lambda^* = \frac{\beta_{c_1}I_c^*}{S+I_c+R_c} + \frac{\beta_{c_2}B_c}{B_c+\kappa_c}.$$

Using (Equation 10), we have a quadratic equation in B_c of the form

$$\nu_2 B_c^2 + \nu_1 B_c + \nu_0 = 0,$$

where

$$\begin{aligned} \nu_2 &= g_c\left[q_c\left(\lambda^*+\mu\right)\left(\mu+\rho_c\right) - \epsilon_c\rho_c\lambda^*\right], \\ \nu_1 &= -\mu_c\kappa_c(\mathcal{R}_b-1)\nu_2, \quad \nu_0 = -\lambda^*\Lambda\kappa_c\alpha_c\left(\mu+\rho_c\right), \end{aligned}$$

with

$$\mathcal{R}_b = \frac{g_c}{\mu_c}.$$

Clearly, $\nu_0 < 0,\ \nu_1 < 0$ if $\mathcal{R}_b > 1$. Since

$$B_c = \frac{-\nu_1 \pm \sqrt{\nu_1^2 - 4\nu_2\nu_0}}{2\nu_2}, \tag{11}$$

it follows that if $\nu_2 < 0,\ \mathcal{R}_b > 1$, then it follows from Descartes' rule of signs that B_c has no positive roots, and if $\nu_2 > 0,\ \mathcal{R}_b > 1$, then B_c has only one positive root. We shall call the positive root B_c^+.

Let

$$B = \frac{B_c^+}{B_c^+ + \kappa_c}.$$

Then

$$\lambda_{c_2}^* = \frac{\beta_{c_2}B_c^+}{B_c^+ + \kappa_c} = \beta_{c_2}B.$$

We have an expression for λ^* such that

$$a_2\lambda^{*2} + a_1\lambda^* + a_0 = 0 \tag{12}$$

where,

$$\begin{aligned} a_2 =& \mu + \epsilon_c + \rho_c > 0,\ a_1 = q_c\,(\mu + \rho_c) - (\beta_{c_1}\,(\mu + \rho_c) \\ & + B\beta_{c_2}(\mu + \epsilon_c + \rho_c)), \\ a_0 =& -Bq_c\beta_{c_2}\,(\mu + \rho_c) < 0. \end{aligned}$$

Since

$$\lambda = \frac{-a_1 \pm \sqrt{a_1^2 - 4a_2a_0}}{2a_2},$$

it follows that if $a_1 > 0$, then the polynomial (Equation 12) has a positive root, and if $a_1 < 0$, then the polynomial (Equation 12) has a positive root. So the polynomial (Equation 12) will always have one positive root.

So system (Equation 2) has a unique endemic equilibrium if $\mathcal{R}_b > 1$.

Remark 1. Due to the symmetric structure of the cholera only and typhoid only sub-models, the typhoid only sub-model has similar structural results to those obtained for the cholera only sub-model. To avoid repetition, we have not shown the analysis of the typhoid only sub-model.

2.3. Cholera-typhoid co-infection model

We study the full co-infection model (Equation 1) in this section.

2.3.1. Non-negative trajectories and boundedness

We prove in this subsection that model (Equation 1) has non-negative trajectories.

Theorem 3. *All solutions of the co-infection model (Equation 1) are non-negative if all the initial conditions are non-negative.*

Proof. Define $t_1 = \sup\{t > 0 | S(\tau_1) > 0, I_c(\tau_1) \geq 0, I_t(\tau_1) \geq 0, I_{ct}(\tau_1) \geq 0, R_c(\tau_1) \geq 0, R_t(\tau_1) \geq 0, R_{ct}(\tau_1) \geq 0,$
$B_c(\tau_1) \geq 0, B_t(\tau_1) \geq 0, \forall \tau_1 \in [0, t]\}$. It follows that $t_1 > 0$ since $S_0 > 0, I_{c0} \geq 0, I_{t0} \geq 0, I_{ct0} \geq 0, R_{c0} \geq 0,$
$R_{t0} \geq 0, R_{t0} \geq 0, B_{c0} \geq 0, B_{t0} \geq 0$.
Assume $t_1 < \infty$, then $S(t_1) > 0, I_c(t_1) = 0, I_t(t_1) = 0, I_{ct}(t_1) = 0, R_c(t_1) = 0, R_t(t_1) = 0, R_{ct}(t_1) = 0,$
$B_c(t_1) = 0, B_t(t_1) = 0$. Applying variation of constants to

$$\frac{dS}{dt} = \Lambda - (\lambda_{c_1} + \lambda_{c_2} + \lambda_{t_1} + \lambda_{t_2})S - \mu S + \rho_c R_c + \rho_t R_t + \rho_{ct} R_{ct}$$

yields

$$S(t_1) = \int_0^{t_1} f(r) \exp\left(-\int_r^{t_1} P(x)dx\right) dr$$

$$+S_0 \exp\left(-\int_0^{t_1} P(x)dx\right),$$

Where $P(x) = \lambda_{c_1} + \lambda_{c_2} + \lambda_{t_1} + \lambda_{t_2} + \mu$ and $f(r) = \Lambda + \rho_c R_c + \rho_t R_t + \rho_{ct} R_{ct}$. Clearly,

$$S(t_1) > 0$$

since $f(r) > 0$ and $P(x) > 0$ when $x, r \in [0, t_1]$. Similarly, $I_c(t_1) > 0, I_t(t_1) > 0, I_{ct}(t_1) > 0, R_t(t_1) > 0$ and $R_c(t_1) > 0$. This produces a contradiction, hence $t_1 = \infty$.

Theorem 4. *All solutions of the co-infection model (Equation 1) are bounded within Ω whenever $g_c \geq \mu_c$ and $g_t \geq \mu_t$.*

Proof. Since $\delta_{ct}(I_c + I_{ct}) \geq 0$, it follows that the upper bound for the time derivative of the total human population, $N(t)$, is

$$\frac{dN}{dt} = \Lambda - \mu N - \delta_{ct}(I_c + I_{ct}) \leq \Lambda - \mu N.$$

Using separation of variables, we obtain the following upper bound for the human population,

$$N \leq \frac{\Lambda - M\exp(-\mu t)}{\mu} \leq \frac{\Lambda}{\mu}.$$

This upper bound for the population implies that each of the classes are also bounded above by the same constant Λ/μ. Since $I_c, I_{ct} \leq \Lambda/\mu$, it follows that the upper bound for the bacterial concentration of *Vibros Cholerae* is bounded above by

$$\begin{aligned} \frac{dB_c}{dt} =& g_c B_c\left(1 - \frac{B_c}{k_c}\right) + \alpha_c I_c + \theta_c I_{ct} - \mu_c B_c \\ \leq& g_c B_c\left(1 - \frac{B_c}{k_c}\right) + (\alpha_c + \theta_c)\frac{\Lambda}{\mu} - \mu_c B_c. \end{aligned} \tag{13}$$

From inequality (Equation 13), if

$$B_c \geq (\alpha_c + \theta_c)\frac{\Lambda}{\mu}, \tag{14}$$

then

$$\frac{dB_c}{dt} \leq (g_c - \mu_c)B_c - \frac{g_c}{k_c}B_c^2 + B_c$$

$$= (g_c - \mu_c + 1)B_c\left(1 - \frac{g_c B_c}{k_c(g_c - \mu_c + 1)}\right). \tag{15}$$

The constant

$$\frac{k_c(g_c - \mu_c + 1)}{g_c}, \tag{16}$$

is the upper bound for the differential inequality (Equation 15) since (Equation 15) is the logistic growth model with carrying capacity (Equation 16). For some $t \geq 0$, $(\alpha_c + \theta_c)\Lambda/\mu$ is an upper bound for B_c whenever (Equation 14) is false, whilst B_c is bounded above by Equation (16) for the rest of the time points in the domain of B_c if (Equation 14) is true. The constant $(\alpha_c +$

$\theta_c)\dfrac{\Lambda}{\mu}$ is the maximum shedding rate from the cholera infected individuals and dually infected individuals. In both cases, $B_c \leq \max\left\{\dfrac{k_c(g_c-\mu_c+1)}{g_c}, (\alpha_c+\theta_c)\dfrac{\Lambda}{\mu}\right\}$.

It is observed that the disease free equilibria, X_1, X_2, X_3, are always unstable due to the condition, $g_c \geq \mu_c$, that is requires for their existence.

The Jacobian of the full system is

$$J(\mathbf{X}_0) = \begin{pmatrix} -\mu & -\beta_{c_1} & -\beta_{t_1} & -(\beta_{c_1}\eta_c+\beta_{t_1}\eta_t) & \rho_c & \rho_t & \rho_{ct} & -\frac{\Lambda\beta_{c_2}}{\mu\kappa_c} & -\frac{\Lambda\beta_{t_2}}{\mu\kappa_t} \\ 0 & \beta_{c_1}-q_c & 0 & \eta_c\beta_{c_1} & 0 & 0 & 0 & \frac{\Lambda\beta_{c_2}}{\mu\kappa_c} & 0 \\ 0 & 0 & \beta_{t_1}-q_t & \eta_t\beta_{t_1} & 0 & 0 & 0 & 0 & \frac{\Lambda\beta_{t_2}}{\mu\kappa_t} \\ 0 & 0 & 0 & -(\mu+\delta_{ct}+\epsilon_{ct}) & 0 & 0 & 0 & 0 & 0 \\ 0 & \epsilon_c & 0 & 0 & -(\mu+\rho_c) & 0 & 0 & 0 & 0 \\ 0 & 0 & \epsilon_t & 0 & 0 & -(\mu+\rho_t) & 0 & 0 & 0 \\ 0 & 0 & 0 & \epsilon_{ct} & 0 & 0 & -(\mu+\rho_{ct}) & 0 & 0 \\ 0 & \alpha_c & 0 & \theta_c & 0 & 0 & 0 & g_c-\mu_c & 0 \\ 0 & 0 & \alpha_t & \theta_t & 0 & 0 & 0 & 0 & g_t-\mu_t \end{pmatrix}, \tag{17}$$

Within the feasible region,

$$\begin{aligned} \Omega_{co} = \Bigg\{&(S, I_c, I_t, I_{ct}, R_c, R_t, R_{ct}, B_c, B_t) \in \mathbb{R}^9_+ \Bigg| 0 \leq N \leq \frac{\Lambda}{\mu}, \\ &B_c \in \left[0, \max\left\{\frac{k_c(g_c-\mu_c+1)}{g_c}, (\alpha_c+\theta_c)\frac{\Lambda}{\mu}\right\}\right], \\ &B_t \in \left[0, \max\left\{\frac{k_t(g_t-\mu_t+1)}{g_t}, (\alpha_t+\theta_t)\frac{\Lambda}{\mu}\right\}\right]\Bigg\}, \end{aligned}$$

We have summarized the results on the boundedness and positivity of the solutions to the co-infection model (Equation 1).

2.3.2. Stability analysis of the disease free equilibrium and reproduction number, $\mathcal{R}_0$

We find the conditions required for the disease free equilibrium for dynamical system (Equation 1) to be locally asymptotically stable in this section. The disease free equilibria of dynamical system (Equation 1) are

$$\begin{aligned} \mathbf{X}_0 =& (S, I_c, I_t, I_{ct}, R_c, R_t, R_{ct}, B_c, B_t) = \left(\frac{\Lambda}{\mu}, 0, 0, 0, 0, 0, 0, 0, 0\right), \\ \mathbf{X}_1 =& (S, I_c, I_t, I_{ct}, R_c, R_t, R_{ct}, B_c, B_t) = (c_3, 0, 0, 0, 0, 0, 0, c_4, c_5), \\ \mathbf{X}_2 =& (S, I_c, I_t, I_{ct}, R_c, R_t, R_{ct}, B_c, B_t) = (c_6, 0, 0, 0, 0, 0, 0, c_4, 0), \\ \mathbf{X}_3 =& (S, I_c, I_t, I_{ct}, R_c, R_t, R_{ct}, B_c, B_t) = (c_7, 0, 0, 0, 0, 0, 0, 0, c_5), \end{aligned}$$

where

$$c_3 = \frac{\Lambda\left((g_c-\mu_c)k_c+g_c\kappa_c\right)\left((g_t-\mu_t)k_t+g_t\kappa_t\right)}{\mu\left((g_c-\mu_c)k_c+g_c\kappa_c\right)\left((g_t-\mu_t)k_t+g_t\kappa_t\right)+\beta_{c_2}g_t(g_c-\mu_c)k_c+\beta_{t_2}g_c(g_t-\mu_t)k_t},$$

$$\begin{aligned} c_4 =& \frac{g_c-\mu_c}{g_c}k_c, \quad c_5 = \frac{g_t-\mu_t}{g_t}k_t, \\ c_6 =& \frac{\Lambda\left((g_c-\mu_c)k_c+g_c\kappa_c\right)\left(g_t\kappa_t\right)}{\mu\left((g_c-\mu_c)k_c+g_c\kappa_c\right)\left(g_t\kappa_t\right)+\beta_{c_2}g_t(g_c-\mu_c)k_c}, \\ c_7 =& \frac{\Lambda\left(g_c\kappa_c\right)\left((g_t-\mu_t)k_t+g_t\kappa_t\right)}{\mu\left(g_c\kappa_c\right)\left((g_t-\mu_t)k_t+g_t\kappa_t\right)+\beta_{t_2}g_c(g_t-\mu_t)k_t}. \end{aligned}$$

The dynamical system (Equation 1) is locally asymptotically stable if all nine of its eigenvalues have negative real parts. Five of the eigenvalues for the Jacobian, J, are $\lambda_1 = -\mu$, $\lambda_2 = -(\mu+\rho_c)$, $\lambda_3 = -(\mu+\rho_t)$, $\lambda_4 = -(\mu+\rho_{ct})$ and $\lambda_5 = -(\mu+\delta_{ct}+\epsilon_{ct})$. The other four eigenvalues for J are the eigenvalues from the sub-matrix

$$\bar{J} = \begin{pmatrix} \beta_{c_1}-q_c & 0 & \frac{\Lambda\beta_{c_2}}{\mu\kappa_c} & 0 \\ 0 & \beta_{t_1}-q_t & 0 & \frac{\Lambda\beta_{t_2}}{\mu\kappa_t} \\ \alpha_c & 0 & g_c-\mu_c & 0 \\ 0 & \alpha_t & 0 & g_t-\mu_t \end{pmatrix}.$$

The characteristic equation for matrix $\bar{J}$ is $(\lambda^2+\nu_1\lambda+\nu_2)(\lambda^2+\nu_3\lambda+\nu_4)$, where

$$\begin{aligned} \nu_1 &= -\left((g_c-\mu_c)+(\beta_{c_1}-q_c)\right), \\ \nu_2 &= (\beta_{c_1}-q_c)(g_c-\mu_c)\left(1-\mathcal{R}_C\right), \\ \nu_3 &= -\left((g_t-\mu_t)+(\beta_{t_1}-q_t)\right), \\ \nu_4 &= (\beta_{t_1}-q_t)(g_t-\mu_t)\left(1-\mathcal{R}_T\right), \end{aligned}$$

and

$$\begin{aligned} \mathcal{R}_C &= \frac{\alpha_c\beta_{c_2}\Lambda}{\kappa_c\mu\mu_c q_c(1-\mathcal{R}^c_h)(1-\mathcal{R}^c_b)}, \\ \mathcal{R}_T &= \frac{\alpha_t\beta_{t_2}\Lambda}{\kappa_t\mu\mu_t q_t(1-\mathcal{R}^t_h)(1-\mathcal{R}^t_b)}, \\ \mathcal{R}^c_h &= \frac{\beta_{c_1}}{q_c} \quad \mathcal{R}^c_b = \frac{g_c}{\mu_c}, \quad \mathcal{R}^t_h = \frac{\beta_{t_1}}{q_t} \quad \mathcal{R}^t_b = \frac{g_t}{\mu_t}. \end{aligned}$$

The constants $\mathcal{R}^c_b$ and $\mathcal{R}^c_h$ are *the bacterial regeneration threshold* and *the human-to-human sub reproduction number*, respectively, for the cholera only sub-model. The constants $\mathcal{R}^t_b$ and $\mathcal{R}^t_h$ are *the bacterial regeneration threshold* and *the human-to-human sub reproduction number*, respectively, for the typhoid only sub-model. The constants $\mathcal{R}_C$ and $\mathcal{R}_T$ are the so-called *basic reproduction numbers* for the cholera only sub-model and the typhoid only sub-model, respectively. Clearly, $\mathcal{R}^c_h, \mathcal{R}^c_b < 1$ or $\mathcal{R}^c_h, \mathcal{R}^c_b > 1$ if and only if $\mathcal{R}_C > 0$; Similarly, $\mathcal{R}^t_h, \mathcal{R}^t_b < 1$ or $\mathcal{R}^t_h, \mathcal{R}^t_b > 1$ if and only if $\mathcal{R}_T > 0$.

TABLE 1 Parameter values used for numerical simulation.

Par.	Range	Point value	Source	Par.	Range	Point value	Source
β_{c_1}		1	Assumed	g_t		0.014	[21]
β_{t_1}		1	[7]	α_c		10	Assumed
β_{c_2}	(0.1—1)	1.97×10^{-11}	[11–14]	α_t		10	[21]
β_{t_2}		1.97×10^{-11}	[15]	μ	(0.017—0.123)	0.02	[14, 22, 23]
β_{c_3}		0.5	Assumed	μ_t		0.0345	[21]
β_{t_3}		1	Assumed	Λ	(100—467)	449.32	[24]
β_{c_4}		10^{-1}	Assumed	μ_c		0.0345	Assumed
β_{t_4}		10^{-1}	Assumed	ϵ_c	(0.07—0.245)	0.07	[14, 16, 19, 25]
k_c	(10^6—10^9)	5×10^6	[11]	ϵ_t		0.1	[26, 27]
k_t		5×10^6	Assumed	ϵ_{ct}		0.1	Assumed
δ_c		6.58×10^{-1}	[14, 16, 17]	κ_c		0.62	Assumed
δ_t		0.6	[15]	κ_t		0.62	Assumed
ρ_c		8.12×10^{-3}	[18, 19]	θ_c		0.8	Assumed
ρ_t		1.3×10^{-3}	[20]	θ_t		0.8	Assumed
ρ_{ct}		1.3×10^{-3}	Assumed	η_c		7×10^{-4}	Assumed
g_c		0.014	Assumed	η_c		7×10^{-2}	Assumed

We note that

$$\nu_2 > (\beta_{c_1} - q_c)(g_c - \mu_c)\left(1 - \max\{\mathcal{R}_C, \mathcal{R}_T\}\right),$$
$$\nu_4 > (\beta_{t_1} - q_t)(g_t - \mu_t)\left(1 - \max\{\mathcal{R}_C, \mathcal{R}_T\}\right).$$

Thus

$$\mathcal{R}_0 = \max\{\mathcal{R}_C, \mathcal{R}_T\}.$$

The constant $\mathcal{R}_0$ is the *basic reproduction number* for the systems (Equation 1). It follows from the Routh Hurwitz criterion that the four eigenvalues of $\bar{J}$ have negative real parts if $\nu_1, \nu_2, \nu_3, \nu_4 > 0$. It is easy to see that $\nu_1, \nu_2, \nu_3, \nu_4 > 0$ if $\mathcal{R}_h^t < 1, \mathcal{R}_b^t < 1, \mathcal{R}_h^t < 1, \mathcal{R}_b^t < 1$ and $\mathcal{R}_0 < 1$. Hence, a positive basic reproduction number for system (Equation 1) that is less than unity implies that the system is locally asymptotically stable at the disease free equilibrium.

2.3.3. Impact analysis

In this section, we show how cholera affects typhoid, and through symmetry, we show how typhoid affects cholera.

The reproduction numbers for cholera and typhoid are

$$\mathcal{R}_C = \frac{\alpha_c \beta_{c_2} \Lambda}{\kappa_c \mu \mu_c q_c (1 - \mathcal{R}_h^c)(1 - \mathcal{R}_b^c)},$$
$$\mathcal{R}_T = \frac{\alpha_t \beta_{t_2} \Lambda}{\kappa_t \mu \mu_t q_t (1 - \mathcal{R}_h^t)(1 - \mathcal{R}_b^t)}, \tag{18}$$

respectively. These two reproduction numbers are dependent on each other. The constant, Λ/μ, allows for the expression of one reproduction number in terms of the other. From the second equation above, Equation (18), isolating, Λ/μ, yields

$$\mathcal{R}_C = \mathcal{R}_T \frac{\alpha_c \beta_{c_2} \kappa_t \mu_t q_t (1 - \mathcal{R}_h^t)(1 - \mathcal{R}_b^t)}{\alpha_t \beta_{t_2} \kappa_c \mu_c q_c (1 - \mathcal{R}_h^c)(1 - \mathcal{R}_b^c)}. \tag{19}$$

Differentiating $\mathcal{R}_C$ with respect to $\mathcal{R}_T$ gives

$$\frac{\partial \mathcal{R}_C}{\partial \mathcal{R}_T} = \frac{\alpha_c \beta_{c_2} \kappa_t \mu_t q_t (1 - \mathcal{R}_h^t)(1 - \mathcal{R}_b^t)}{\alpha_t \beta_{t_2} \kappa_c \mu_c q_c (1 - \mathcal{R}_h^c)(1 - \mathcal{R}_b^c)}. \tag{20}$$

We conclude that an increase in cholera cases may be associated with an increase in typhoid cases, and an increase in typhoid cases may be associated with an increase in cholera cases. This conclusion is subject to the following conditions: firstly, the *the bacterial regeneration threshold* for both cholera and typhoid must be less than unity; secondly, *the human-to-human sub reproduction number* for both cholera and typhoid must also be less than unity. This result proves the symbiotic nature of the relationship between the typhoid disease and the Cholera disease.

3. Numerical simulations

In this section, we give a brief outline of the numerical results obtained in the investigation. Table 1 shows the parameters of the cholera typhoid co-infection model (Equation 1). The basic reproduction number, $\mathcal{R}_0$, obtained from the Table 1 is 1.4. The initial conditions used to produce the figures in this section were: $S(0) = 99980$, $I_c(0) = 20$, $I_t(0) = 20$, $I_{ct}(0) = 20$, $R_c(0) = 0$, $R_t(0) = 0$, $R_{ct}(0) = 0$, $B_c(0) = 40000$, $B_t(0) = 40000$. Note

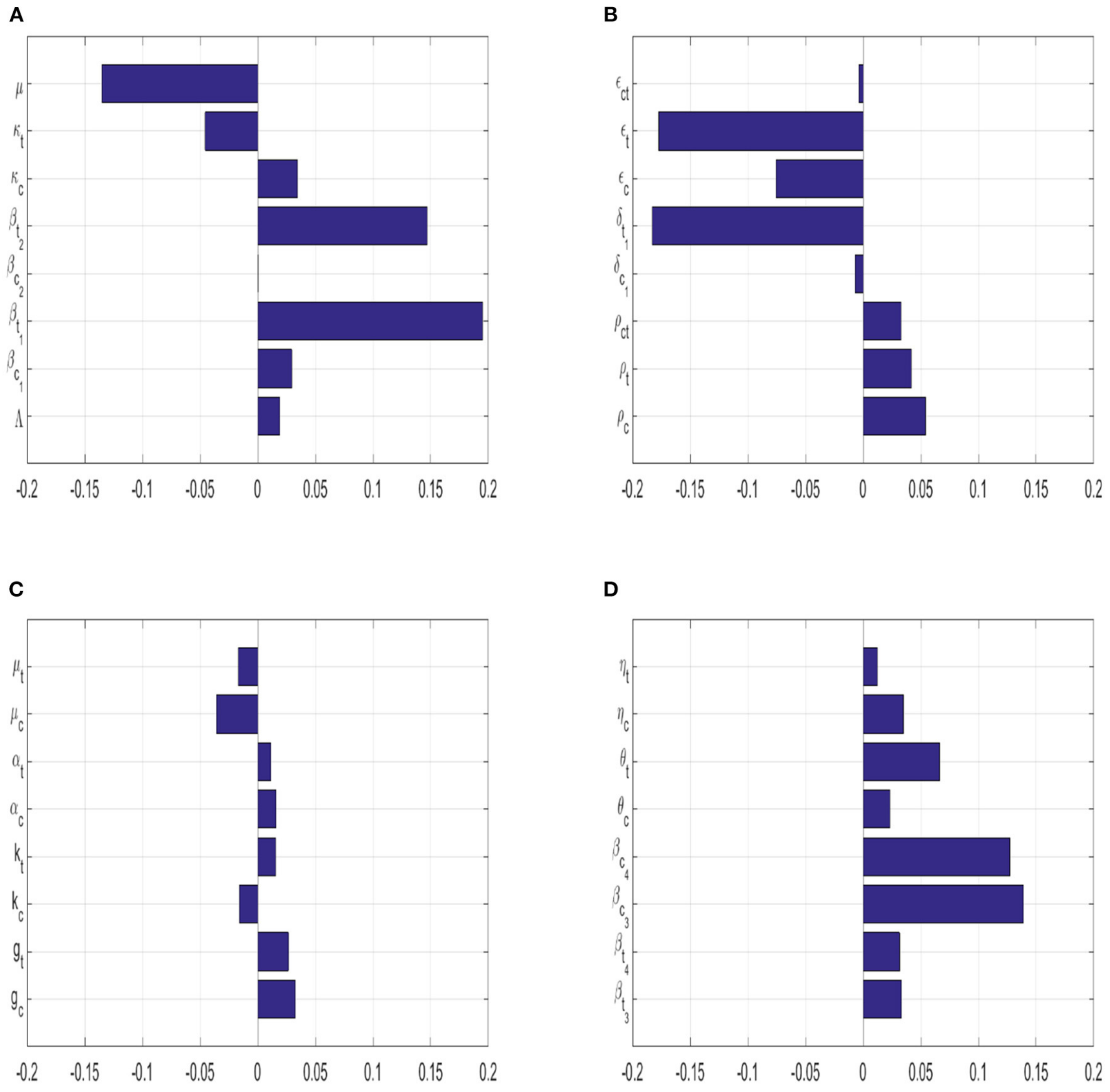


FIGURE 2
The correlation between the co-infected class and each of the model's parameter are shown in this bar graph (PRCC). **(A)** Shows the PRCC values for $\{\Lambda, \beta_{c_1}, \beta_{t_1}, \beta_{c_2}, \beta_{t_2}, \kappa_c, \kappa_t, \mu\}$. **(B)** Shows the PRCC values for $\{\rho_c, \rho_t, \rho_{ct}, \delta_c, \delta_t, \epsilon_c, \epsilon_t, \epsilon_{ct}\}$. **(C)** Shows the PRCC values for $\{g_c, g_t, k_c, k_t, \alpha_c, \alpha_t, \mu_c, \mu_t\}$. **(D)** Shows the PRCC values for $\{\beta_{t_3}, \beta_{t_4}, \beta_{c_3}, \beta_{c_4}, \theta_c, \theta_t, \eta_c, \eta_t\}$.

that all figures in this section are presented in the logarithmic scale since the range of some of the plots spanned several orders of magnitude.

Coupled with the parameters from Table 1, the sensitivity indices of the variables above are shown on Figure 2. Latin Hypercube sampling was utilized to generate the plot above (Figure 2). This method returns the correlation between the state variable I_{ct} and each of the model parameters, and it also returns the ranks of all these correlations (PRCC). The simulation was carried out over 1,000 runs. A parameter with a negative PRCC value means that parameter is negatively correlated with I_{ct}, whilst a parameter with a positive PRCC value represents a positive correlation between that parameter and I_{ct}. Relative to the current model parameters, we note that the coinfection class is most sensitive to changes to the person-to-person typhoid transmission rate, β_{t_1}, and the correlation is positive between this parameter and the state variable. The typhoid induced death rate is that second most sensitive parameter to the coinfection class, and it is negatively correlated to the coinfection class. Due to the large number of parameters in model 1, we have opted to split the PRCC values into 4 equal sets, see Figure 2.

The contour map of $\mathcal{R}_0$ as a function of the typhoid recovery rate, ϵ_t, and the cholera recovery rate, ϵ_c is shown in Figure 3. Using the parameters from Table 1, the base case

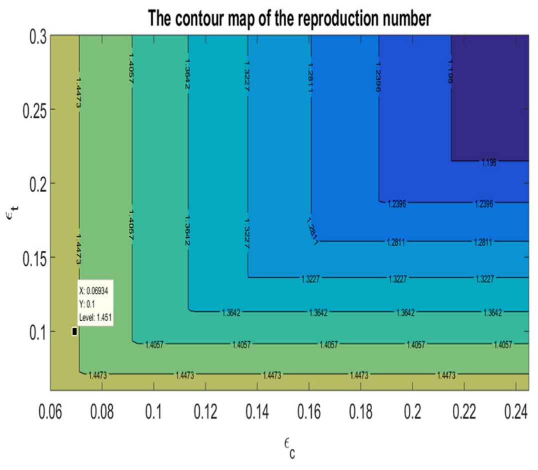


FIGURE 3
The contour map of the basic reproduction number, $\mathcal{R}_0$, as a function of the typhoid recovery rate, ϵ_t, and the cholera recovery rate, ϵ_c.

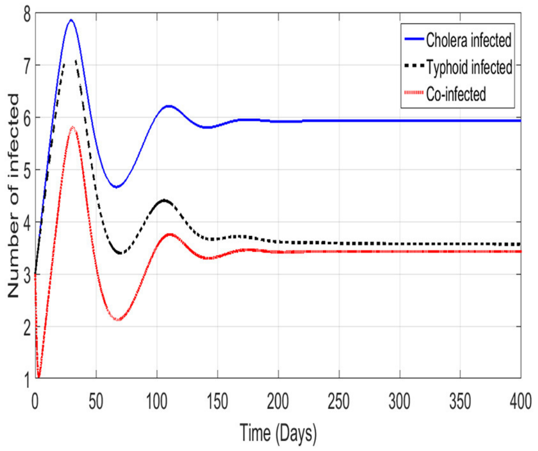


FIGURE 4
The trajectories of the infectious classes.

as well as the contour levels are also shown in Figure 3. The basic reproduction number $\mathcal{R}_0$ attains its global minimum if both the typhoid and cholera recovery rate are maximized. It is be observed that, locally, a reduction in the reproduction number, $\mathcal{R}_0$, —moving the base case to a lower contour level—is only achieved by increasing the cholera recovery rate. Since the reproduction number, $\mathcal{R}_0$, is the maximum of the reproduction numbers of the individual diseases, it follows that a reduction in the reproduction number, $\mathcal{R}_0$, means a reduction in the reproduction numbers of each of the diseases. Hence, locally, an increase in the cholera recovery rate will not only reduce the cholera reproduction number, $\mathcal{R}_c$, but it has the added benefit of indirectly reducing the reproduction number for typhoid, $\mathcal{R}_t$, as well. It is also observed that increasing the typhoid recovery rate exclusively will have no immediate benefits locally. This finding is consistent with the previous findings of an optimal treatment plan being centered around the recovery rate of cholera.

We show the trajectories of the three infectious classes of model (Equation 1). An initial surge in infections followed closely by an immediate recovery is shown in Figure 4. The phenomenon of waning immunity results in the smaller second wave of infections. The co-infected class is the only exception to this observation. We see the co-infected class reach a local minimum before the first surge in cholera only or typhoid only infections is reached. A possible reason for this is that, unlike the cholera and typhoid classes, the co-infected class does not recruit directly from the susceptible class. This is due to the fact that the cholera disease has a shorter incubation period than the typhoid disease. The incubation periods are 1.4 days for cholera [2] and 19 days for typhoid [3]. What is then observed in the co-infected class is a case of people leaving the class either through death or recovery coupled with the delayed recruitment into the class. All the diseases reach stability after the second waves of infection.

In order to understand how the diseases interact with each other, we vary the different recovery rates and observe how the prevalence of each of the infections change. In Figures 5A,B show the impact of varying the recovery rate of the co-infected on the cholera and typhoid prevalence, whilst (Figures 5C,D) show the impact of varying the recovery rates of cholera and typhoid on the prevalence of the co-infected individuals. Figure 5C shows a significant reduction in the co-infected class' prevalence when the cholera recovery rate is increased, whilst plots (Figure 5D) shows that this reduction is negligible when the typhoid recovery rate was increased. Figures 5A,B show that an increase in the co-infected class' recovery rate reduces the typhoid prevalence more than the cholera prevalence. The net effect is that an increased cholera recovery rate may be associated with a decreased prevalence of the co-infected individuals and a higher co-infected recovery rate. This in turn, produces a reduced typhoid prevalence. Given the current model parameters, this finding suggests that an optimal treatment plan for the two infections should primarily focus on increasing the cholera recovery rate as opposed to the typhoid recovery rate. This also underscores the point made earlier about the symbiotic nature of the two diseases.

4. Discussion and conclusion

In this article, we formulated and analyzed a theoretical model for the transmission dynamics of a cholera typhoid co-infection model. Through numerical simulations, we were able to verify a number of the results obtained analytically.

The birth and death rates of the bacteria are central to proving the boundedness and positivity of all three

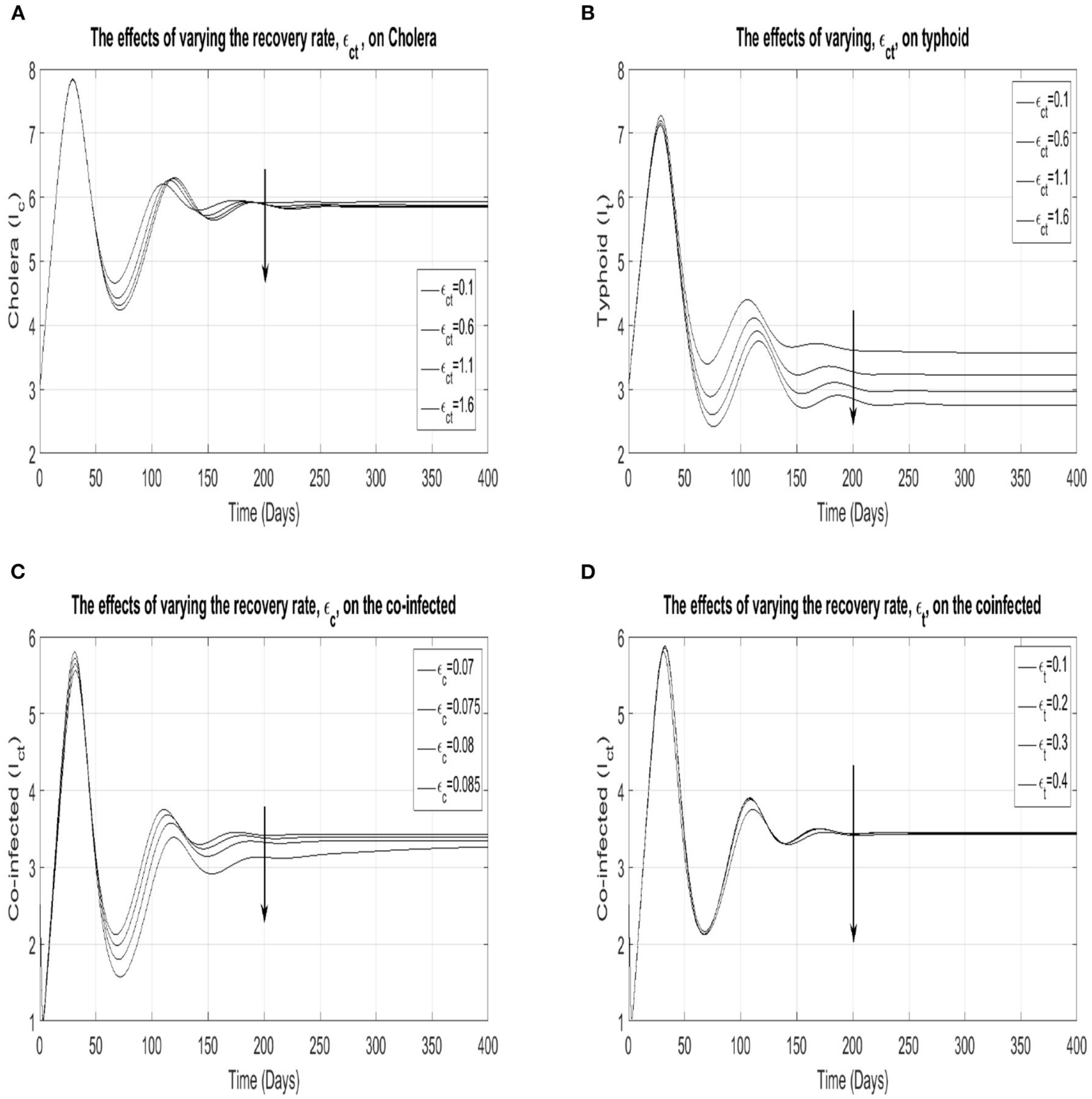


FIGURE 5
Plots **(A,B)** show the cholera and typhoid prevalence, respectively, as the co-infection recovery rate, ϵ_{ct}, runs through {0.1, 0.6, 1.1, 1.6}. Plots **(C,D)** show the prevalence of the co-infected as the cholera and typhoid recovery rates are varied through the sets {0.07, 0.075, 0.08, 0.085} and {0.1, 0.2, 0.3, 0.4}, respectively.

models—cholera only sub-model, typhoid only sub-model, and the full cholera typhoid co-infection model. For the cholera-only model, if the birth rate of the *Vibrio Cholerae* bacteria exceeds its death rate, then the cholera only model has non-negative and bounded trajectories. For the typhoid only model, if the birth rate of the *Salmonella Typhi* bacteria exceeds its death rate, then the typhoid only model has non-negative and bounded trajectories. For the full cholera typhoid co-infection model, if the birth rates of the *Vibrio Cholerae* bacteria and the *Salmonella Typhi* bacteria exceed their death rates, simultaneously, then the cholera-typhoid co-infection model has non-negative and bounded trajectories.

In analyzing the equilibria of the co-infection models, several key sights were discovered. We showed the existence of the disease free equilibria, by finding them, for all three models. Sufficient conditions for the existence of the endemic equilibria for the cholera only sub-model and the typhoid only sub-model were documented. We showed that if the reproduction number is less than one for the all the models, then the disease free equilibria are locally asymptotically stable, otherwise they are unstable. Global stability could not be guaranteed, both at the disease free equilibria and the endemic equilibria, in any of the models. Sensitivity analysis revealed the parameters in the model were at the heart of the spread of the cholera typhoid co-infection. The prevalence of cholera is decreased whenever η_t, β_{t_3}, β_{t_4} are increased and/or β_{c_1}, β_{c_2}, ϵ_{ct}, η_c and θ_c are decreased. The prevalence of typhoid is decreased whenever η_c, β_{c_3}, β_{c_4} are increased and/or β_{t_1}, β_{t_2}, ϵ_{ct}, η_t and θ_t are decreased.

From the impact analysis section, we found that an increase in cholera cases may be associated with an increased risk of typhoid and that an increase in typhoid cases may be associated with an increased risk of cholera. This result proves the symbiotic nature of the relationship between the typhoid disease and the cholera disease.

The findings in this investigation come with some limitations. The most glaring of all is the lack of data to fit the model to. Our model also fails to take into account the highly seasonal nature of each of the diseases. For the two infections, fear has a significant impact on the transmission dynamics. Future work should also be able to account for the effects of fear in the transmission dynamics of both infections. Notwithstanding these limitations, we believe that the findings of this investigation can still be useful to policy makers in containing an outbreak of these two diseases.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

LM prepared the manuscript and contributed to the analysis, whilst FN formulated the model, proofread the manuscript, and also contributed to the analysis. Both authors contributed to the article and approved the submitted version.

Acknowledgments

The authors would like to acknowledge the financial support received from the University of Johannesburg during the production of this manuscript.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Brachman PS, Evans AS. *Bacterial Infections of Humans: Epidemiology and Control*. 3rd ed. Springer (1998). Available online at: http://gen.lib.rus.ec/book/index.php?md5=ae64651f2efa7ade60c537325be29cab

2. Azman AS, Rudolph KE, Cummings DA, Lessler J. The incubation period of cholera: a systematic review. *J Infect*. (2013) 66:432–8. doi: 10.1016/j.jinf.2012.11.013

3. Taylor A, Santiago A, Gonzalez-Cortes A, Gangarosa EJ. Outbreak of typhoid fever in Trinidad in 1971 traced to a commercial ice cream product. *Am J Epidemiol*. (1974) 100:150–7. doi: 10.1093/oxfordjournals.aje.a112017

4. Akinyi OC, Mugisha J, Manyonge A, Ouma C, Maseno K. Modelling the impact of misdiagnosis and treatment on the dynamics of malaria concurrent and co-infection with pneumonia. *Appl Math Sci*. (2013) 7:6275–96. doi: 10.12988/ams.2013.39521

5. Onyinge DO, Ongati NO, Odundo F. Mathematical model for co-infection of HIV/AIDS and pneumonia with treatment. *Int J Scientific Eng Appl Sci*. (2016) 2:106–11. Available online at: http://ir.jooust.ac.ke:8080/xmlui/handle/123456789/2643

6. Mushayabasa S, Bhunu CP, Mhlanga NA. Modeling the transmission dynamics of typhoid in malaria endemic settings. *Appl Appl Math*. (2014) 9:121–40. Available online at: https://digitalcommons.pvamu.edu/aam/vol9/iss1/9

7. Mushayabasa S, Bhunu CP, Ngarakana-Gwasira ET. Assessing the impact of drug resistance on the transmission dynamics of typhoid fever. *Comput Biol J*. (2013) 2013:1–13. doi: 10.1155/2013/303645

8. Pitzer VE, Bowles CC, Baker S, Kang G, Balaji V, Farrar JJ, et al. Predicting the impact of vaccination on the transmission dynamics of typhoid in South Asia: A mathematical modeling study. *PLoS Negl Trop Dis*. (2014) 8:e2642. doi: 10.1371/journal.pntd.0002642

9. Khan MA, Parvez M, Islam S, Khan I, Shafie S, Gul T. Mathematical analysis of typhoid model with saturated incidence rate. *Adv Stud Biol*. (2015) 7:65–78. doi: 10.12988/asb.2015.41059

10. Matsebula L, Nyabadza F, Mushanyu J. Mathematical analysis of typhoid fever transmission dynamics with seasonality and fear. *Commun Math Biol Neurosci*. (2021) 2021:36. doi: 10.28919/cmbn/5590

11. Codeço CT. Endemic and epidemic dynamics of cholera: the role of the aquatic reservoir. *BMC Infect Dis*. (2001) 1:1. doi: 10.1186/1471-2334-1-1

12. Bertuzzo E, Casagrandi R, Gatto M, Rodriguez-Iturbe I, Rinaldo A. On spatially explicit models of cholera epidemics. *J R Soc Interface*. (2010) 7:321–33. doi: 10.1098/rsif.2009.0204

13. Mari L, Bertuzzo E, Righetto L, Casagrandi R, Gatto M, Rodriguez-Iturbe I, Rinaldo A. Modelling cholera epidemics: the role of waterways, human mobility and sanitation. *J R Soc Interface*. (2012) 9:376–88. doi: 10.1098/rsif.2011.0304

14. Miller Neilan RL, Schaefer E, Gaff H, Fister KR, Lenhart S. Modeling optimal intervention strategies for cholera. *Bull Math Biol*. (2010) 72:2004–18. doi: 10.1007/s11538-010-9521-8

15. Mushayabasa S. Impact of vaccines on controlling typhoid fever in Kassena-Nankana district of upper east region of Ghana: insights from a mathematical model. *J Modern Math Stat*. (2011) 5:54–9. doi: 10.3923/jmmstat.2011.54.59

16. Sepulveda J, Gomez-Dantes H, Bronfman M. Cholera in the Americas: an overview. *Infection*. (1992) 20:243–8. doi: 10.1007/BF01710787

17. Shuai Z, Tien JH, van den Driessche P. Cholera models with hyperinfectivity and temporary immunity. *Bull Math Biol*. (2012) 74:2423–45. doi: 10.1007/s11538-012-9759-4

18. King AA, Ionides EL, Pascual M, Bouma MJ. Inapparent infections and cholera dynamics. *Nature*. (2008) 454:877–80. doi: 10.1038/nature07084

19. Sanches RP, Ferreira CP, Kraenkel RA. The role of immunity and seasonality in cholera epidemics. *Bull Math Biol*. (2011) 73:2916–31. doi: 10.1007/s11538-011-9652-6

20. Okosun K, Makinde OD. Modelling the impact of drug resistance in malaria transmission and its optimal control analysis. *Int J Phys Sci*. (2011) 6:6479–87. doi: 10.5897/IJPS10.542

21. Mutua JM, Wang FB, Vaidya NK. Modeling malaria and typhoid fever co-infection dynamics. *Math Biosci*. (2015) 264:128–44. doi: 10.1016/j.mbs.2015.03.014

22. Hartley DM, Morris Jr JG, Smith DL. Hyperinfectivity: a critical element in the ability of *V. cholerae* to cause epidemics? *PLoS Med*. (2006) 3:e7. doi: 10.1371/journal.pmed.0030007

23. Munro PM, Colwell RR. Fate of *Vibrio cholerae* O1 in seawater microcosms. *Water Res*. (1996) 30:47–50. doi: 10.1016/0043-1354(95)00137-A

24. Blayneh K, Cao Y, Kwon HD. Optimal control of vector-borne diseases: treatment and prevention. *Discrete Continuous Dyn Syst B*. (2009) 11:587. doi: 10.3934/dcdsb.2009.11.587

25. Hendrix TR. The pathophysiology of cholera. *Bull N Y Acad Med*. (1971) 47:1169–80.

26. Adetunde IA. Mathematical models for the dynamics of typhoid fever in Kassena-Nankana district of upper east region of Ghana. *J Modern Math Stat*. (2008) 2:45–9. Available online at: https://medwelljournals.com/abstract/?doi=jmmstat.2008.45.49

27. Mushayabasa S. Modeling the impact of optimal screening on typhoid dynamics. *Int J Dyn Control*. (2016) 4:330–8. doi: 10.1007/s40435-014-0123-4

frontiers | Frontiers in Applied Mathematics and Statistics

TYPE Original Research
PUBLISHED 28 September 2022
DOI 10.3389/fams.2022.993152

Check for updates

OPEN ACCESS

EDITED BY
Yong Chen,
Hangzhou Dianzi University, China

REVIEWED BY
P. Senthil Kumar,
MVJ College of Engineering, India
Wenchang Luo,
Ningbo University, China

*CORRESPONDENCE
Moi Hua Tuh
tuhmoihua@uitm.edu.my

SPECIALTY SECTION
This article was submitted to Optimization,
a section of the journal
Frontiers in Applied Mathematics and Statistics

RECEIVED 13 July 2022
ACCEPTED 07 September 2022
PUBLISHED 28 September 2022

CITATION
Tuh MH, Kon CML, Chua HS and Lau MF (2022) Optimal statistical design of the double sampling *np* chart based on expected median run length. *Front. Appl. Math. Stat.* 8:993152. doi: 10.3389/fams.2022.993152

Optimal statistical design of the double sampling *np* chart based on expected median run length

Moi Hua Tuh[1,2]*, Cynthia Mui Lian Kon[2], Hong Siang Chua[2] and Man Fai Lau[3]

[1]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Kota Samarahan, Malaysia, [2]Faculty of Engineering, Computing and Science, Swinburne University of Technology Sarawak Campus, Kuching, Malaysia, [3]School of Software and Electrical Engineering, Swinburne University of Technology, Melbourne, VIC, Australia

Double sampling (DS) control charts are widely regarded as an effective process monitoring tool owing to their remarkable properties, such as the ability to detect small and moderate process shifts efficiently with the reduced sample size. Since the shape of the run length distribution is highly right-skewed for the process small shift size and becomes almost symmetric when the process shift size is large, the use of median run length (MRL) as a performance measure is therefore more representative. Existing works on the DS *np* chart construction were performed by taking an approach that the shift size of the process fraction nonconforming is assumed to be known. However, the shift size of the fraction nonconforming is usually unknown by the quality practitioners in practice. Herein, to address this issue, the expected median run length (EMRL) has been suggested as a performance measure for the unknown shift size. This paper suggests an optimal design procedure for the DS *np* chart based on the EMRL criterion. An example is provided to illustrate the construction of the EMRL-based DS *np* chart. The DS *np* chart is compared with a competing chart based on the EMRL criterion. Findings obtained reveal that when the shift size is unknown, the EMRL is an alternative performance measure for the DS *np* chart, with greater sensitivity observed for the DS *np* chart in contrast to the standard *np* chart for detecting a wide range of shifts.

KEYWORDS
median run length, unknown shift size, fraction nonconforming, numerical integration, standard *np* chart

Introduction

Control chart is one of the most useful tools in Statistical Process Control since control charts play a key role in detecting the assignable cause(s) [1]. Other effective way to mitigate the incidence of false alarm rate and to increase the control chart sensitivity includes the fuzzy logic scheme [2–6], which combines the probability and fuzzy set theories for enabling inference of process state based on fuzzified sensitivity criteria. When the quality characteristics can only be classified into two possible outcomes, for instance, "Yes or No," "Good or Bad," "Conforming or Nonconforming," and "Defective or Non-defective," it is not possible to monitor the process using the variable control charts, such as the $\bar{X}$, s, and R charts. In such a scenario, attribute control charts will be the right choice.

The standard *np* chart is one of the attribute control charts that has been widely used for process monitoring. Compared to the *p* chart, the standard *np* chart is also easier to understand by managers who are lack of statistical knowledge and new to the quality control system. This provides more persuasive evidence of quality issues to management [7]. However, the standard *np* chart is well known to be slow in detecting moderate and small process fraction nonconforming (p) shifts. Consequently, considerable attentions have been devoted to develop *np* chart with various approaches for enhancing the sensitivity of the standard *np* chart in the literature, such as the optimal design for the cumulative sum (CUSUM) *np* chart by Gan [8] and the modified exponentially weighted moving average (EWMA) *np* chart by Gan [9]. Adaptive technique to develop *np* control chart has also been studied. Case in point, Epprecht and Costa [10] investigated the *np* properties for sample size that fluctuates between small and large sizes, while Luo and Wu [11] proposed optimal designs of variable sample size and variable sampling intervals *np* charts under steady-state mode.

Croasdale [12] was the first to introduce the DS scheme, bringing the concept of DS process from the acceptance sampling field and applying the technique to the $\bar{X}$ chart. Following Croasdale [12], Daudin [13] demonstrated that by employing the sample size of n_1 at stage 1 and combining two samples of size n_1 and n_2 at stage 2 can improve the performance of the $\bar{X}$ chart and this reduces the number of items to be inspected, resulting in a cost-saving benefit in the manufacturing process. As a result, the DS scheme developed after 1992, such as He and Grigoryan [14], Costa and Claro [15], Torng and Lee [16], Khoo et al. [17], and De Araujo Rodrigues et al. [18], to name a few, were based on the method proposed by Daudin [13]. De Araujo Rodrigues et al. [18] were the first to introduce the DS *np* chart. Chong et al. [19], Joekes et al. [20], Lee and Khoo [21], and Tuh et al. [22] have since focused their studies around the proposed DS *np* chart.

The performance of the control charts is usually evaluated by the average run length (ARL). ARL is defined as the average number of samples to be plotted on the control chart before the out-of-control signal is observed. However, many researchers criticized the sole dependence of the ARL as the performance measure of control charts, for example, see Teoh et al. [23], Khoo et al. [24], Lee and Khoo [25], Smajdorová and Noskievičová [26]. In addition, as pointed out by Graham et al. [27], the ARL as a performance measure has many drawbacks. It is noted that the run length (RL) distribution is changing from highly right-skewed when process shift size is small to almost symmetric when process shift size is large. Consequently, utilizing the ARL as a performance measure may neglect some vital statistical properties of control charts. Chakraborti [28] recommended to investigate the percentiles of the run length distribution such as 5, 25, 50 (median), 75, and 95th percentiles to have a better vision and evaluation of the RL distribution. Utilizing the median run length (MRL) that is the 50th percentile of the RL has some additional benefits in designing control charts [29–31]. This is due to the fact that the MRL is less impacted by the skewness of the RL distribution. Thus, the MRL provides a more accurate measure of the central tendency compared to the ARL [32].

Existing work on the DS *np* control chart based on MRL by Tuh et al. [22] assumes the shift size is known. However, the shift size of the process fraction nonconforming is usually unknown by quality practitioners. The performance of control charts may be negatively impacted if the determined shift size differs from the actual value. To overcome this issue, it is crucial to consider the expected median run length (EMRL) as an alternative performance measure, where only a range of process shift sizes is required. You et al. [33], Teoh et al. [34], Tang et al. [35], Chong et al. [36], and Yeong et al. [37], to name a few, evaluated the performance of control charts when the process shift size is unknown. Motivated by these studies, we suggest the optimal design of the DS *np* chart based on EMRL in this paper.

The paper is structured as follows: Section Theories and formulations begins with a brief introduction of the standard *np* and DS *np* charts, followed by a discussion of the RL distribution properties of the DS *np* chart. Section Computational methods and results presents the optimization design of the EMRL-based DS *np* chart, performance of the DS *np* chart, and comparison to that of the standard *np* chart. The operability of the DS *np* chart is also furnished through an illustrative example incorporating event within a data processing department. Finally, the conclusion is given in section Conclusions.

Theories and formulations

The standard *np* chart

The goal of the standard *np* chart is to detect the assignable causes for increasing shift in the process fraction nonconforming. As a result, the standard *np* chart is designed without a lower control limit. According to Lee and Khoo [29], the probability that $d <$ UCL is calculated as follows:

$$A_S = P\left(d \leq \lfloor UCL \rfloor\right) = \sum_{d=0}^{\lfloor UCL \rfloor} \frac{n!}{d!\left(n-d\right)!} p^d \left(1-p\right)^{n-d} \quad (1)$$

where $p = p_0$ when $\gamma = 1$, and $p = p_1$ when $\gamma \neq 1$. The d and UCL represent the number of nonconforming items found in a sample of size n and upper control limit of the standard *np* chart, respectively.

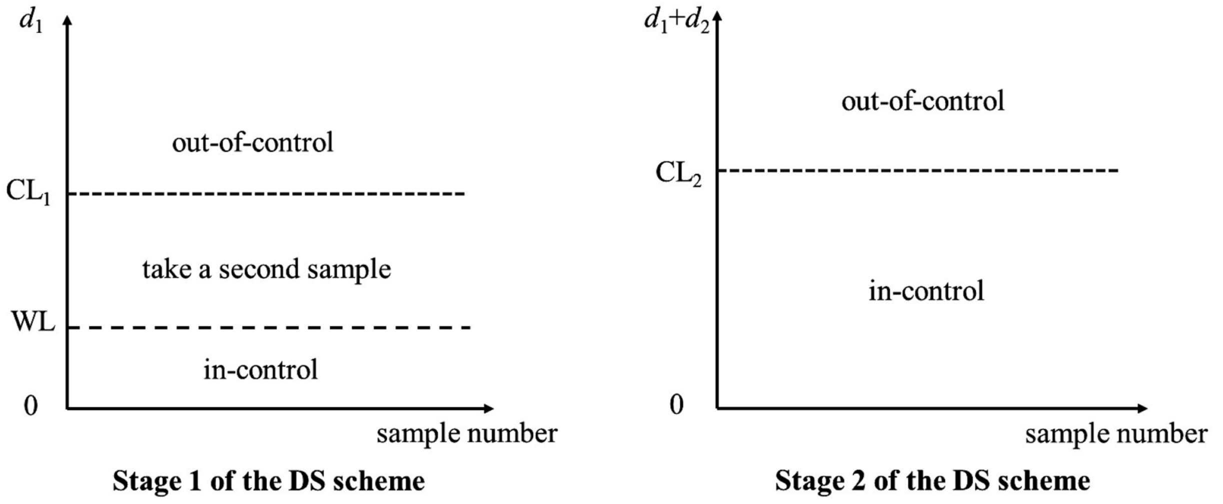


FIGURE 1
Regions at stages 1 and 2 of the DS schemes.

The DS *np* chart

In this section, we give a brief review of the DS *np* chart, which was first introduced by De Araujo Rodrigues [18]. To achieve the desired statistical performance, the DS *np* chart is designed with five charting parameters. We define the set of charting parameters as n_1, n_2, WL, CL_1, and CL_2, where n_1, n_2, WL, CL_1, and CL_2 denote the size of the first sample, the size of second sample, the stage 1 warning limit, the stage 1 control limit, and the stage 2 control limit, respectively. The three non-integer control limits are set as $\mathrm{WL} = \mathrm{Ac}_1 + 0.5$, $\mathrm{CL}_1 = \mathrm{Re} – 0.5$, and $\mathrm{CL}_2 = \mathrm{Ac}_2 + 0.5$ to avoid doubt by quality practitioners when the number of nonconforming items in a sample falls within or outside the control limits. In these expressions, Ac_1, Re_1, and Ac_2 are the acceptance number in the first sample, the rejection number in the first sample, and the acceptance number in the stage 2, respectively. The operation of the DS *np* chart is elaborated in the following steps. The graphical summary is shown in Figure 1.

Step 1. Determine the limits that are WL, $\mathrm{CL}_{1,}$ and CL_2.
Step 2. Take the first sample of size n_1 from the process and check the number of nonconforming items (d_1).
Step 3. At the stage 1 of the DS scheme,

a) if $d_1 < \mathrm{WL}$, the process is considered as in-control and return to Step 2.
b) if $d_1 > \mathrm{CL}_1$, the process is considered as out-of-control. For the purpose of identifying and eliminating the assignable cause(s), corrective measure is performed. Repeat Step 2.
c) if $\mathrm{WL} < d_1 < \mathrm{CL}_1$, take a second sample with size n_2. Count the number of nonconforming items (d_2) for the second sample. Then, move to the next step, which is stage 2 of the DS scheme.

Step 4. If $(d_1 + d_2) < \mathrm{CL}_2$, the process is considered to be in-control and return to Step 2. Else, the process is deemed to be out-of-control. To locate and remove the assignable cause(s), corrective action is once again performed. Repeat Step 2.

The run length properties of the DS *np* chart

In general, RL denotes the number of sample points plotted on the DS *np* chart before the first signal is observed. The probability mass function (pmf) $f_{RL}(\zeta)$ and the cumulative distribution function (cdf) $F_{RL}(\zeta)$ of the RL distribution for a control chart are

$$f_{RL}(\zeta) = (1 - A)\, A^{\zeta - 1} \tag{2}$$

and

$$F_{RL}(\zeta) = P(RL \leq \zeta) = 1 - A^{\zeta}, \tag{3}$$

respectively [38], where $\zeta \in \{1, 2, 3, 4, \ldots\}$ and A is calculated by Equations (5) and (6).

As suggested by Chakraborti [28], the smallest integer of the percentile run length, ζ_α, can be obtained from

$$\zeta_\alpha \geq \frac{\ln(1-\alpha)}{\ln A} \tag{4}$$

facilitates the computation for the $100\alpha th$ $(0 < \alpha < 1)$ percentile of the RL.

The probability that the process is in-control is given by $A = A_1 + A_2$. Here, A_1 denotes the probability that $d_1 <$ WL at the stage 1 of the DS scheme, while A_2 is the probability that WL $< d_1 <$ CL$_1$ at the stage 1 of the DS scheme and $(d_1 + d_2) <$ CL$_2$ at the stage 2 of the DS scheme, where

$$A_1 = P\left(d_1 \leq \lfloor WL \rfloor\right) = \sum_{d_1=0}^{\lfloor WL \rfloor} \frac{n_1!}{d_1!\left(n_1-d_1\right)!} p^{d_1}\left(1-p\right)^{n_1-d_1} \tag{5}$$

and

$$A_2 = P\left(\lfloor WL \rfloor < d_1 < \lceil CL_1 \rceil\right) \cap P\left(d_1 + d_2 \leq \lfloor CL_2 \rfloor\right)$$
$$= \sum_{d_1=\lfloor WL \rfloor+1}^{\lceil CL_1 \rceil-1} \left[\frac{n_1!}{d_1!\left(n_1-d_1\right)!} p^{d_1}\left(1-p\right)^{n_1-d_1} \left(\sum_{d_2=0}^{\lfloor CL_2 \rfloor-d_1} \frac{n_2!}{d_2!\left(n_2-d_2\right)!} p^{d_2}\left(1-p\right)^{n_2-d_2}\right)\right], \tag{6}$$

where $\lfloor \cdot \rfloor$ denotes the round down to the nearest integer and $\lceil \cdot \rceil$ represents the round up to the nearest integer.

The efficiency of the DS *np* chart is determined by how fast the chart can detect an increasing shift in the process fraction nonconforming p with the shift size $\gamma = \frac{p_1}{p_0}$, where $p_1 > p_0$. Note that $p = p_0$ and $p = p_1$ for the in-control $(\gamma = 1)$ and out-of-control $(\gamma > 1)$ states, respectively. According to De Araujo Rodrigues et al. [18], the ARL and the average sample size (ASS) can be computed as

$$ARL = \frac{1}{1-A} \; and \tag{7}$$
$$ASS = n_1 + n_2 P_s\,, \tag{8}$$

respectively, where $P_s = P\left(\lfloor WL \rfloor < d_1 < \lceil CL_1 \rceil\right)$. The in-control ARL (ARL$_0$) and ASS (ASS$_0$) are calculated when $p = p_0$, while the out-of-control ARL (ARL$_1$) and ASS (ASS$_1$) can be obtained when $p = p_1$.

The MRL is the RL with a cumulative probability of at least 50% of the time. The MRL can be computed using Equation (4) by putting $\alpha = 0.5$, where Equation (4) can be rewritten as

$$\zeta_{0.5} \geq \frac{\ln(0.5)}{\ln A}, \tag{9}$$

where $\zeta_{0.5}$ = MRL. Note that MRL = MRL$_0$ is the in-control MRL when $\gamma = 1$, whereas MRL = MRL$_1$ is the out-of-control MRL when $\gamma > 1$.

The computation of the percentiles of the RL requires the shift size to be known in advance. However, in practical, it is usually tough for practitioners to quantify the magnitude of process shift due to insufficient historical data. Aside from that, the shift size varies according to various undetermined or random events [39]. Thus, the percentile of the RL can be replaced by the expected percentile of the RL $(E(\zeta_\alpha))$. Herein, a specific value for γ is not required and can be determined as follows:

$$E(\zeta_\alpha) = \int_{\gamma_{\min}}^{\gamma_{\max}} f_\gamma\ (\gamma)\zeta_\alpha(\gamma)\, d\gamma. \tag{10}$$

Hence, the expected median run length, EMRL, that is $E(\zeta_{0.5})$ can be computed as

$$EMRL = E(\zeta_{0.5}) = \int_{\gamma_{\min}}^{\gamma_{\max}} f_\gamma\ (\gamma) MRL(\gamma)\, d\gamma. \tag{11}$$

In this paper, the EMRL in Equation (11) is evaluated by using a numerical integration over the probability density function $f_\gamma(\gamma)$ for a shift size interval of $\gamma_{\min}$ (the lower limit of the integral) to $\gamma_{\max}$ (the upper limit of the integral). The function $f_\gamma(\gamma)$ is assumed to have a continuous uniform distribution over the interval $(\gamma_{\min}, \gamma_{\max})$ [39], with probability density function of $f_\gamma(\gamma) = \frac{1}{(\gamma_{\max}-\gamma_{\min})}$, where $\gamma_{\max} - \gamma_{\min}$ denotes the interval length. To incorporate exact shift sizes, $\gamma \in \{1.5, 2.0, 3.0\}$, that were considered in Tuh et al. [22], two intervals of the shift size, therefore, are set in this paper: (i) $(\gamma_{\min}, \gamma_{\max}] = (1.1, 2.0]$ and (ii) $(\gamma_{\min}, \gamma_{\max}] = (2.0, 3.0]$. For example, the interval $(\gamma_{\min}, \gamma_{\max}] = (1.1, 2.0]$ and $(\gamma_{\min}, \gamma_{\max}] = (2.0, 3.0]$ include $\gamma = \{1.5, 2.0\}$ and $\gamma = \{3.0\}$, respectively. Note that MRL(γ) denotes the MRL$_1$ at γ. The Gauss Legendre Quadrature is employed to estimate approximately the definite integral in Equation (11).

This paper also evaluates the expected average run length (EARL) and the expected average sample size (EASS) values through

$$EARL = \int_{\gamma_{\min}}^{\gamma_{\max}} f_\gamma\ (\gamma) ARL(\gamma)\, d\gamma \tag{12}$$

and

$$EASS = \int_{\gamma_{\min}}^{\gamma_{\max}} f_\gamma\ (\gamma) ASS(\gamma)\, d\gamma, \tag{13}$$

respectively.

Computational methods and results

Optimal design of the EMRL-based DS *np* chart

Tuh et al. [22] investigated the performance of the DS *np* chart using the MRL as the performance measure. Interested readers may refer to Tuh et al. [22] for the detailed optimization procedure for the DS *np* chart based on the MRL.

Nevertheless, the actual process shift size is usually unknown. Thus, the DS *np* chart can be designed for a given range of shift sizes $(\gamma_{\min},\ \gamma_{\max}]$, which is an alternative method. The optimization design of the DS *np* chart by minimizing the out-of-control expected median run length (EMRL_1) is given as

$$\min_{n1,n2,WL,CL1,CL2} EMRL_1 \tag{14}$$

subject to:

$$EMRL_0 \geq MRL_{0min} \ and \tag{15}$$

$$EASS_0 = n. \tag{16}$$

$\mathrm{MRL}_{0\mathrm{min}}$ [in Constraint (15)] and n [in Constraint (16)] are denoted as the predetermined in-control median run length and predetermined in-control average sample size, respectively, where $n_1 < n < n_2$, with both n_1 and n_2 are integers. Note that $\mathrm{EMRL}_0 = \mathrm{MRL}_0$ and $\mathrm{EASS}_0 = \mathrm{ASS}_0$ are considered in this paper.

The procedure for searching optimal (n_1, n_2, WL, CL_1, CL_2) combination, based on the optimization model in (14)–(16), and the DS *np* chart based on EMRL is outlined as follows:

Step 1: Specify the desired values of p_0, *n*, $\mathrm{MRL}_{0\mathrm{min}}$, $\gamma_{\min}$, and $\gamma_{\max}$. Here, n is the average sample size in each sampling when the process is in a state of control; n is also the fixed sample size for the standard *np* chart.

Step 2: Initialize $\mathrm{EMRL}_{1\mathrm{min}}$ with a very large value, say 10^5. $\mathrm{EMRL}_{1\mathrm{min}}$ is used to keep track of the lowest EMRL_1 value.

Step 3: Begin with $n_1 = 1$.

Step 4: With the current n_1 value, determine the combination of (n_1, $n_{2,}$ WL, CL_1) for a specified n when $\gamma = 1$, such that the Constraint (16) is fulfilled. The value of n_2 is computed through the rearrangement of Equation (8), that is, $n_2 = {}^{(n-n_1)}\!/_{\mathrm{P}(\lfloor \mathrm{WL} \rfloor < d_1 < \lceil \mathrm{CL}_1 \rceil)}$, and is rounded up to the nearest integer, where $0 < WL < \ CL_1$.

Step 5: Then determine CL_2 based on the Equation (9) and Constraint (15), in which the computed EMRL equals to EMRL_0 when $\gamma = 1$, where $\mathrm{CL}_2 > \mathrm{CL}_1$. The values of WL, CL_1, and CL_2 are determined based on operating procedure discussed in Section 2.2. In this step, the possible (n_1, n_2, WL, CL_1, CL_2) combination is identified.

Step 6: Once the possible (n_1, n_2, WL, CL_1, CL_2) combination has been determined, EMRL_1 will be computed for $p = p_1$, by means of Equation (11). If the calculated EMRL_1 is less than the current $\mathrm{EMRL}_{1\mathrm{min}}$, the $\mathrm{EMRL}_{1\mathrm{min}}$ value will be replaced by the newly computed EMRL_1. The current (n_1, n_2, WL, CL_1, CL_2) combination is temporarily stored as the possible combination before any new lower EMRL_1 value is found. If the (n_1, n_2, WL, CL_1, CL_2) combination obtained in the following search yields similar $\mathrm{EMRL}_{1\mathrm{min}}$, the combination will be saved together as a possible combination. Otherwise, the (n_1, n_2, WL, CL_1, CL_2) combination will not be considered if it results in larger EMRL_1 value.

Step 7: Once the search with $n_1 = 1$ is complete, increase n_1 by one. Repeat Steps 4–6, for the remaining $n_1 = 2$, $3\ldots$, $(n - 1)$, to search for the possible (n_1, n_2, WL, CL_1, CL_2) combinations that satisfy the Constraints (15)–(16) and having the smallest value of EMRL_1.

Step 8: If more than one combinations of (n_1, n_2, WL, CL_1, CL_2) produce a similar lowest EMRL_1 value, the combination that yields the smallest out-of-control expected average sample size (EASS_1) value is selected as the optimal combination.

An optimization MATLAB program is developed to execute the above procedure to search for the optimal (n_1, n_2, WL, CL_1, CL_2) combination for the EMRL-based DS *np* chart.

In this paper, based on the Gauss Legendre Quadrature rule, the weights (w_i) and nodes (x_i) values are obtainable through the MATLAB coding written by Winckel [40]. These values are considered for the computation of $E(\zeta_\alpha)_1$, $\mathrm{EARL}_{1,}$ and EASS_1. According to Hale and Townsend [41], the fundamental accuracy can be achieved for any number of ordinates (N) that exceeds 100. Therefore, N = 200 is considered for all these computations.

Comparative studies

In this section, the EMRL_1 performance of the standard *np* chart with unknown shift size is compared with that of the DS *np* chart. The computational procedure for the standard *np* chart based on the EMRL_1 is to find the minimal value of UCL given the sample size n, by attaining the constraint $EMRL_0 \geq MRL_{0min}$. The $E(\zeta_{0.5})_0\ (= MRL_0)$ and $EARL_0\ (= ARL_0)$ of the standard *np* chart are computed using Equations (9) and (7), respectively, by replacing A with A_S from Equation (1). The optimal charting parameters of the DS *np* chart are computed using the optimization procedure described in Section Optimal design of the EMRL-based DS *np* chart. The different combinations of input parameters as follows are considered: $p_0 \in \{0.005,\ 0.01,\ 0.02\}$, $MRL_{0min} \in \{200,\ 370.4\}$, $n \in \{25,\ 50,\ 100,\ 200,\ 400,\ 800\}$, and two intervals of process shift sizes: (1) $(\gamma_{\min}, \gamma_{\max}] = (1.1,\ 2.0]$ and (2) $(\gamma_{\min}\ ,\ \gamma_{\max}] = (2.0\ ,\ 3.0]$. We only provide

TABLE 1 Charting parameters with corresponding $(E(\zeta_{0.05})_0$, EMRL$_0$, $E(\zeta_{0.95})_0)$ and EARL$_0$ for standard *np* and optimal DS *np* charts, when MRL$_{0\text{min}}$ = 200.

			Standard *np* chart				DS *np* chart		
					Expected Percentile RL			Expected Percentile RL	
γ_{min}	γ_{max}	p_0	n	UCL	$(E(\zeta_{0.05})_0, EMRL_0, E(\zeta_{0.95})_0)$	EARL$_0$	$(n_1, n_2$, WL, CL$_1$, CL$_2)$	$(E(\zeta_{0.05})_0, EMRL_0, E(\zeta_{0.95})_0)$	EARL$_0$
1.1	2.0	0.005	100	3.5	(31, 414, 1,789)	597.63	(8, 2,340, 0.5, 2.5, 17.5)	(16, 205, 882)	294.82
			200	5.5	(91, 1,229, 5,311)	1773.23	(57, 4,298, 1.5, 4.5, 29.5)	(15, 200, 861)	287.76
			400	7.5	(49, 658, 2,841)	948.59	(151, 6,111, 2.5, 6.5, 41.5)	(15, 200, 864)	288.73
			800	10.5	(19, 251, 1,084)	362.20	(293, 8,333, 3.5, 8.5, 56.5)	(15, 200, 863)	288.52
		0.01	50	3.5	(33, 434, 1,876)	626.50	(24, 1,090, 1.5, 4.5, 16.5)	(15, 203, 876)	292.60
			100	4.5	(15, 202, 872)	291.35	(33, 1,557, 1.5, 4.5, 23.5)	(16, 213, 920)	307.44
			200	7.5	(51, 685, 2,958)	987.60	(80, 2,578, 2.5, 6.5, 36.5)	(15, 200, 864)	288.80
			400	10.5	(20, 258, 1,116)	372.71	(144, 4,459, 3.5, 9.5, 59.5)	(15, 200, 862)	287.94
		0.02	25	3.5	(36, 480, 2,071)	691.62	(2, 580, 0.5, 2.5, 17.5)	(17, 221, 952)	318.03
			50	4.5	(16, 216, 932)	311.55	(17, 740, 1.5, 4.5, 22.5)	(15, 201, 868)	289.95
			100	7.5	(56, 744, 3,214)	1,073.03	(39, 1,427, 2.5, 5.5, 39.5)	(16, 206, 891)	297.69
			200	10.5	(21, 274, 1,183)	395.16	(101, 1,882, 4.5, 9.5, 52.5)	(15, 201, 866)	289.25
2.0	3.0	0.005	100	3.5	(31, 414, 1,789)	597.63	(23, 708, 0.5, 2.5, 8.5)	(15, 200, 862)	288.00
			200	5.5	(91, 1,229, 5,311)	1,773.23	(99, 1,143, 1.5, 5.5, 12.5)	(16, 206, 888)	296.84
			400	7.5	(49, 658, 2,841)	948.59	(233, 1,506, 2.5, 7.5, 16.5)	(15, 202, 870)	290.61
			800	10.5	(19, 251, 1,084)	362.20	(528, 2,135, 4.5, 9.5, 23.5)	(16, 213, 920)	307.47
		0.01	50	3.5	(33, 434, 1,876)	626.50	(34, 352, 1.5, 4.5, 8.5)	(15, 202, 873)	291.78
			100	4.5	(15, 202, 872)	291.35	(47, 658, 1.5, 5.5, 13.5)	(16, 206, 887)	296.29
			200	7.5	(51, 685, 2,958)	987.60	(116, 756, 2.5, 7.5, 16.5)	(15, 200, 864)	288.60
			400	10.5	(20, 258, 1,116)	372.71	(268, 994, 4.5, 9.5, 22.5)	(15, 200, 865)	288.96
		0.02	25	3.5	(36, 480, 2,071)	691.62	(7, 136, 0.5, 2.5, 7.5)	(17, 218, 940)	314.11
			50	4.5	(16, 216, 932)	311.55	(25, 282, 1.5, 4.5, 12.5)	(17, 224, 967)	323.19
			100	7.5	(56, 744, 3,214)	1,073.03	(60, 336, 2.5, 7.5, 15.5)	(16, 208, 897)	299.89
			200	10.5	(21, 274, 1,183)	395.16	(134, 500, 4.5, 9.5, 22.5)	(15, 201, 868)	290.14

TABLE 2 Charting parameters with corresponding and EARL_0 for standard *np* and optimal DS *np* charts, when $\text{MRL}_{0\text{min}} = 370.4$.

			Standard *np* chart				DS *np* chart		
					Expected Percentile RL			Expected Percentile RL	
γ_{min}	γ_{max}	p_0	n	UCL	$(E(\zeta_{0.05})_0, EMRL_0, E(\zeta_{0.95})_0)$	EARL_0	$(n_1, n_2, \text{WL}, \text{CL}_1, \text{CL}_2)$	$(E(\zeta_{0.05})_0, EMRL_0, E(\zeta_{0.95})_0)$	EARL_0
1.1	2.0	0.005	100	3.5	(31, 414, 1,789)	597.63	(38, 3,985, 1.5, 3.5, 27.5)	(30, 393, 1,697)	566.84
			200	5.5	(91, 1,229, 5,311)	1,773.23	(59, 3,979, 1.5, 4.5, 29.5)	(28, 375, 1,617)	540.25
			400	7.5	(49, 658, 2,841)	948.59	(144, 7,069, 2.5, 7.5, 48.5)	(28, 372, 1,606)	536.36
			800	11.5	(59, 786, 3,396)	1,133.91	(374, 10,324, 4.5, 10.5, 69.5)	(28, 372, 1,605)	536.07
		0.01	50	3.5	(33, 434, 1,876)	626.50	(23, 1,230, 1.5, 3.5, 19.5)	(30, 393, 1,696)	566.43
			100	5.5	(96, 1,297, 5,603)	1,870.79	(27, 2,454, 1.5, 4.5, 34.5)	(29, 385, 1,661)	554.77
			200	7.5	(51, 685, 2,958)	987.60	(66, 4,670, 2.5, 6.5, 60.5)	(29, 382, 1,650)	551.07
			400	11.5	(61, 816, 3,526)	1,177.46	(189, 4,974, 4.5, 10.5, 67.5)	(28, 375, 1,621)	541.40
		0.02	25	3.5	(36, 480, 2,071)	691.62	(11, 719, 1.5, 3.5, 21.5)	(28, 371, 1,602)	535.00
			50	5.5	(108, 1,450, 6,263)	2,091.10	(13, 1,373, 1.5, 4.5, 37.5)	(29, 383, 1,652)	551.81
			100	7.5	(56, 744, 3,214)	1,073.03	(37, 1,679, 2.5, 8.5, 46.5)	(28, 371, 1,600)	534.37
			200	11.5	(66, 882, 3,810)	1,272.00	(93, 2,728, 4.5, 9.5, 72.5)	(28, 373, 1,609)	537.32
2.0	3.0	0.005	100	3.5	(31, 414, 1,789)	597.63	(58, 1,223, 1.5, 4.5, 12.5)	(29, 389, 1,679)	560.71
			200	5.5	(91, 1,229, 5,311)	1,773.23	(98, 1,175, 1.5, 5.5, 13.5)	(28, 378, 1,631)	544.67
			400	7.5	(49, 658, 2,841)	948.59	(143, 1,599, 1.5, 5.5, 17.5)	(29, 381, 1,646)	549.64
			800	11.5	(59, 786, 3,396)	1,133.91	(497, 2,850, 4.5, 11.5, 28.5)	(28, 372, 1,608)	537.15
		0.01	50	3.5	(33, 434, 1,876)	626.50	(32, 442, 1.5, 4.5, 10.5)	(31, 416, 1,794)	599.25
			100	5.5	(96, 1,297, 5,603)	1,870.79	(49, 590, 1.5, 5.5, 13.5)	(28, 376, 1,623)	542.01
			200	7.5	(51, 685, 2,958)	987.60	(116, 756, 2.5, 7.5, 17.5)	(30, 399, 1,724)	575.89
			400	11.5	(61, 816, 3,526)	1,177.46	(252, 1,340, 4.5, 10.5, 27.5)	(28, 372, 1,605)	536.00
		0.02	25	3.5	(36, 480, 2,071)	691.62	(16, 225, 1.5, 4.5, 10.5)	(30, 399, 1,722)	575.27
			50	5.5	(108, 1,450, 6,263)	2,091.10	(26, 253, 1.5, 4.5, 12.5)	(29, 387, 1,671)	558.17
			100	7.5	(56, 744, 3,214)	1,073.03	(58, 381, 2.5, 7.5, 17.5)	(30, 395, 1,706)	569.92
			200	11.5	(66, 882, 3,810)	1,272.00	(126, 675, 4.5, 12.5, 27.5)	(28, 371, 1,602)	535.05

the results for the combinations of n and p_0 such that $np_0 = \{0.5, 1.0, 2.0, 4.0\}$. The and p_0 combinations that generate $np_0 = \{0.5, 1.0, 2.0, 4.0\}$ were also adopted by several researchers [see [10], [29], and [19]] for clarity and unbiased comparison between competing charts.

Performance of the standard *np* and DS *np* charts based on EMRL

The charting parameter UCL for the standard np and the optimal charting parameters $(n_1, n_2, \text{WL}, \text{CL}_1, \text{CL}_2)$ of the DS np chart based on the EMRL_1 are listed in Tables 1, 2. The corresponding values of $E(\zeta_{0.05})_0$, EMRL_0, $E(\zeta_{0.95})_0$, and EARL_0 are also provided in the tables. Note that $E(\zeta_{0.05})_0$ and $E(\zeta_{0.95})_0$ denote the in-control 5th and 95th percentiles of the RL, respectively. For example, Table 2 shows that when $p_0 = 0.01, n = 100$ and $(\gamma_{\min}, \gamma_{\max}] = (1.1, \quad 2.0]$, for the standard np chart, while $(n_1, n_2, \text{WL}, \text{CL}_1, \text{CL}_2) = (27, 2454, 1.5, 4.5, 34.5)$ for the optimal DS np chart. The DS np chart with these charting parameters gives the smallest EMRL_1 value, while the EMRL_0 is at least 370.4. Subsequently, the corresponding values of $(E(\zeta_{0.05})_0, \text{EMRL}_0, E(\zeta_{0.95})_0, \text{EARL}_0)$ for the standard np and optimal DS np charts are computed as (96, 1,297, 5,603, 1,870.79) and (29, 385, 1,661, 554.77), respectively. The optimal design makes the DS np chart easier to implement in practice. Consider the case of a plastic component created *via* injection molding, for which a rapid detection within the range of process shift sizes $(\gamma_{\min}, \gamma_{\max}] = (1.1, \quad 2.0]$ is required. Table 1 suggests $(n_1, n_2, \text{WL}, \text{CL}_1, \text{CL}_2) = (24, 1{,}090, 1.5, 4.5, 16.5)$ as the best charting parameter for detecting this range of shift sizes if $p_0 = 0.01$, $n = 50$, and $\text{MRL}_{0\min} = 200$.

In Table 3, the $E(\zeta_{0.05})_1$, EMRL_1, $E(\zeta_{0.95})_1$, and EARL_1 values, for the out-of-control case, can be obtained using the charting parameter UCL for the standard np and optimal charting parameters $(n_1, n_2, \text{WL}, \text{CL}_1, \text{CL}_2)$ of the DS np charts (refer to Tables 1, 2). For instance, when $p_0 = 0.02$, $n = 50$, $\text{MRL}_{0\min} = 200$, and $(\gamma_{\min}, \gamma_{\max}] = (1.1, \quad 2.0]$, Table 1 gives $(n_1, n_2, \text{WL}, \text{CL}_1, \text{CL}_2) = (17, 740, 1.5, 4.5, 22.5)$ as the optimal charting parameters for the DS np chart. With these optimal charting parameters, $(E(\zeta_{0.05})_1, \text{EMRL}_1, E(\zeta_{0.95})_1, \text{EARL}_1) = (1.83, 18.50, 78.34, 26.49)$. The equations used for the evaluation of $E(\zeta_{0.05})_1$, EMRL_1, $E(\zeta_{0.95})_1$, and EARL_1 values can be found in Section The run length properties of the DS np chart.

Numerical results in Tables 1, 2 clearly demonstrate that the EMRL_0 values are lower than EARL_0, for both standard np and optimal DS np charts for the in-control case ($\gamma = 1$). For instance, referring to Table 1, the DS np chart gives $\text{EARL}_0 = 314.11$ when $p_0 = 0.02, = 25$, $(\gamma_{\min}, \gamma_{\max}] = (2.0, \quad 3.0]$, and $\text{MRL}_{0\min} = 200$. Practitioners may interpret a false alarm happens by the 314th sample in half of the time. In fact, this value is located in between 60 and 70th (= 378) percentile of the RL distribution, and the false alarm actually happens before 314th sample, that is by the 218th sample ($\text{EMRL}_0 = 218$), occurs in half of the time. On the contrary, for the out-of-control case (see Table 3), when $p_0 = 0.02$, $n = 100$, $(\gamma_{\min}, \gamma_{\max}] = (2.0, \quad 3.0]$, and $\text{MRL}_{0\min} = 200$, the DS np chart gives $\text{EARL}_1 = 2.06$, while $\text{EMRL}_1 = 1.44$, showing small difference between the EARL_1 and EMRL_1 values. This demonstrates that when the RL distribution is highly right-skewed, the average is significantly larger than the median. In contrast, the average is relatively closer to the median in symmetric distribution. Consequently, we recommend the EMRL over EARL as a performance measure which delivers a clearer interpretation for the performance DS np chart. In addition, based on the EMRL performance measure, Table 3 shows that the optimal DS np chart outperforms the standard np chart for all shift sizes, $(\gamma_{\min}, \gamma_{\max}]$, with the former giving lower EMRL_1 than the latter for identical p_0, n, $\text{MRL}_{0\min}$, and $(\gamma_{\min}, \gamma_{\max}]$ combination.

Performance of the standard *np* and DS *np* charts based on expected percentile of the RL distribution

The percentiles of RL distribution can help to reveal more information about the entire RL distribution, including the early false alarm rates. In this paper, the $E(\zeta_{0.05})$ and $E(\zeta_{0.95})$ are also analyzed to equip practitioners with a better view on the spread of the entire RL distribution of the standard np and optimal DS np charts.

The lower percentile, such as $E(\zeta_{0.05})$ evaluated in this paper for the in-control case $(\gamma = 1)$, provides information concerning early false alarm rates. Let us consider standard np chart in Table 1, when $p_0 = 0.01$, $n = 50$, $MRL_{0min} = 200$, and $(\gamma_{\min}, \gamma_{\max}] = (1.1, \quad 2.0]$, gives $E(\zeta_{0.05})_0 = 33$. This result suggests a false alarm will occur by 33rd sample point in 5% of the time. On the contrary, a false alarm will happen in half of the time by the 434th sample $\left(E(\zeta_{0.5})_0 = 434\right)$, meaning that sample 434 has a chance of 0.5 of detecting a false alarm, whereas the EARL_0 is indicated as 626.50.

On the other hand, the higher percentile of the RL distribution, for example, $E(\zeta_{0.95})_1$, provides information about the out-of-control condition which will be issued by the control chart with a high possibility at a certain magnitude of the shift. Based on DS np chart, as shown in Table 3, when $p_0 = 0.005$, $MRL_{0min} = 370.4$, $n = 100$, and $(\gamma_{\min}, \gamma_{\max}] = (2.0, \quad 3.0]$, this chart is anticipated to signal within the first 20.83 samples with a probability of 0.95 $(E(\zeta_{0.95})_1 = 20.83)$. In other words, practitioners can claim with 95% confidence that an out-of-control signal will be discovered by the 20.83rd sample.

Moreover, both the standard np and DS np charts, shown in Tables 1, 2, clearly demonstrate that the in-control RL is subject to significant variation. Expectedly, in Table 2, the in-control extreme percentile of the DS np chart is 1,722 – 30 = 1,692 and

TABLE 3 Performance of the DS *np* and standard *np* charts with $\text{MRL}_{0\min} = 200$ and 370.4.

				$\text{MRL}_{0min} = 200$				$\text{MRL}_{0min} = 370.4$			
				Standard *np* chart		**DS *np* chart**		**Standard *np* chart**		**DS *np* chart**	
				Expected percentile RL		**Expected percentile RL**		**Expected percentile RL**		**Expected percentile RL**	
$\gamma_{\min}$	$\gamma_{\min}$	p_0	n	$(E(\zeta_{0.05})_1, EMRL_1, E(\zeta_{0.95})_1)$	EARL_1	$(E(\zeta_{0.05})_1, EMRL_1, E(\zeta_{0.95})_1)$	EARL_1	$(E(\zeta_{0.05})_1, EMRL_1, E(\zeta_{0.95})_1)$	EARL_1	$(E(\zeta_{0.05})_1, EMRL_1, E(\zeta_{0.95})_1)$	EARL_1
1.1	2.0	0.005	100	(8.71, 111.50, 480.36)	160.69	(2.45, 27.62, 117.74)	39.65	(8.71, 111.50, 480.36)	160.69	(3.35, 38.73, 165.72)	55.66
			200	(16.11, 211.50, 912.32)	304.86	(1.80, 18.69, 79.09)	26.73	(16.11, 211.50, 912.32)	304.86	(2.27, 24.92, 106.04)	35.74
			400	(6.95, 87.11, 374.84)	125.46	(1.44, 12.42, 51.91)	17.67	(6.95, 87.11, 374.84)	125.46	(1.68, 15.88, 67.01)	22.71
			800	(2.45, 26.60, 113.20)	38.12	(1.22, 7.82, 32.23)	11.10	(5.27, 65.39, 280.74)	94.06	(1.36, 9.85, 40.88)	14.00
		0.01	50	(9.08, 116.26, 500.90)	167.54	(2.47, 27.91, 119.09)	40.08	(9.08, 116.26, 500.90)	167.54	(3.33, 39.35, 168.48)	56.57
			100	(3.94, 46.33, 198.60)	66.63	(1.88, 19.12, 80.96)	27.36	(16.86, 221.47, 955.66)	319.34	(2.24, 24.84, 105.66)	35.61
			200	(7.16, 89.97, 387.16)	129.58	(1.45, 12.33, 51.61)	17.58	(7.16, 89.97, 387.16)	129.58	(1.65, 16.13, 68.13)	23.07
			400	(2.48, 27.09, 115.47)	38.89	(1.22, 7.81, 32.04)	11.05	(5.41, 67.17, 288.87)	96.75	(1.36, 9.86, 41.03)	14.04
		0.02	25	(9.88, 126.99, 547.04)	182.93	(2.51, 28.45, 121.28)	40.82	(9.88, 126.99, 547.04)	182.93	(3.20, 37.50, 106.37)	53.86
			50	(4.12, 48.91, 209.81)	70.38	(1.83, 18.50, 78.34)	26.49	(18.53, 244.00, 1052.93)	351.80	(2.23, 24.64, 105.00)	35.39
			100	(7.60, 96.17, 413.94)	138.52	(1.46, 12.50, 52.26)	17.78	(7.60, 96.17, 413.94)	138.52	(1.68, 15.72, 66.41)	22.51
			200	(2.57, 28.24, 120.35)	40.51	(1.24, 7.82, 32.28)	11.13	(5.70, 71.23, 306.20)	102.54	(1.34, 9.72, 40.41)	13.83
2.0	3.0	0.005	100	(1.96, 20.56, 87.20)	29.44	(1.00, 4.56, 18.15)	6.41	(1.96, 20.56, 87.20)	29.44	(1.00, 5.24, 20.83)	7.30
			200	(1.88, 19.86, 84.05)	28.39	(1.00, 2.59, 9.54)	3.55	(1.88, 19.86, 84.05)	28.39	(1.00, 2.80, 10.48)	3.86
			400	(1.00, 6.30, 25.64)	8.90	(1.00, 1.45, 5.08)	2.09	(1.00, 6.30, 25.64)	8.90	(1.00, 1.67, 5.86)	2.33
			800	(1.00, 1.97, 7.04)	2.73	(1.00, 1.00, 2.74)	1.37	(1.00, 2.88, 10.73)	3.94	(1.00, 1.01, 2.91)	1.42
		0.01	50	(2.01, 21.16, 89.81)	30.32	(1.00, 4.62, 18.41)	6.49	(2.01, 21.16, 89.81)	30.32	(1.00, 5.17, 20.79)	7.28
			100	(1.00, 7.36, 30.15)	10.41	(1.00, 2.60, 9.66)	3.59	(1.93, 20.40, 86.51)	29.21	(1.00, 2.78, 10.38)	3.83
			200	(1.01, 6.38, 25.95)	9.01	(1.00, 1.45, 5.03)	2.08	(1.01, 6.38, 25.95)	9.01	(1.00, 1.57, 5.45)	2.21
			400	(1.00, 1.97, 7.05)	2.74	(1.00, 1.00, 2.71)	1.36	(1.00, 2.90, 10.78)	3.96	(1.00, 1.01, 2.86)	1.41
		0.02	25	(2.13, 22.52, 95.71)	32.28	(1.00, 4.75, 18.92)	6.67	(2.13, 22.52, 95.71)	32.28	(1.00, 5.08, 20.37)	7.15
			50	(1.01, 7.55, 31.03)	10.70	(1.00, 2.58, 9.48)	3.53	(2.02, 21.64, 91.88)	31.01	(1.00, 2.84, 10.58)	3.89
			100	(1.02, 6.55, 26.69)	9.25	(1.00, 1.44, 4.96)	2.06	(1.02, 6.55, 26.69)	9.25	(1.00, 1.54, 5.34)	2.18
			200	(1.00, 1.98, 7.08)	2.74	(1.00, 1.00, 2.67)	1.35	(1.00, 2.92, 10.88)	4.00	(1.00, 1.00, 2.83)	1.40

TABLE 4 MRL$_1$ computed using the optimal charting parameters of the EMRL-based DS *np* chart and the MRL-based DS *np* chart for $p_0 = 0.005$, $n = 100$, and EMRL$_0$ ∈ {200, 370.4}.

			MRL$_1$			
MRL$_{0\min}$	**Type of DS *np* chart**	$(\gamma_{\min}, \gamma_{\max}]$	$\gamma = 1.2$	$\gamma = 1.5$	$\gamma = 2.0$	$\gamma = 3.0$
200	EMRL-based design chart	(1.1, 2.0]	59	20	10	–
		(2.0, 3.0]	–	–	–	3
	MRL-based design chart	–	59	20	8	3
370.4	EMRL-based design chart	(1.1, 2.0]	86	27	13	–
		(2.0, 3.0]	–	–	–	3
	MRL-based design chart	–	82	26	9	3

TABLE 5 MRL$_1$ computed using the optimal charting parameters of the EMRL-based DS *np* chart and the MRL-based DS *np* chart for $p_0 = 0.01$, $n = 100$, and EMRL$_0$ ∈ 200, 370.4.

			MRL$_1$			
MRL$_{0\min}$	**Type of DS *np* chart**	$(\gamma_{\min}, \gamma_{\max}]$	$\gamma = 1.2$	$\gamma = 1.5$	$\gamma = 2.0$	$\gamma = 3.0$
200	EMRL-based design chart	(1.1, 2.0]	45	12	5	–
		(2.0, 3.0]	–	–	–	2
	MRL-based design chart	–	41	12	4	2
370.4	EMRL-based design chart	(1.1, 2.0]	59	15	7	
		(2.0, 3.0]	–	–	–	2
	MRL-based design chart	-	56	15	5	2

TABLE 6 MRL$_1$ computed using the optimal charting parameters of the EMRL-based DS *np* chart and the MRL-based DS *np* chart for $p_0 = 0.02$, $n = 100$, and EMRL$_0$ ∈ {200, 370.4}.

			MRL$_1$			
MRL$_{0\min}$	**Type of DS *np* chart**	$(\gamma_{\min}, \gamma_{\max}]$	$\gamma = 1.2$	$\gamma = 1.5$	$\gamma = 2.0$	$\gamma = 3.0$
200	EMRL-based design chart	(1.1, 2.0]	29	7	4	–
		(2.0, 3.0]	–	–	–	1
	MRL-based design chart	-	28	7	2	1
370.4	EMRL-based design chart	(1.1, 2.0]	38	8	4	–
		(2.0, 3.0]	–	–	–	1
	MRL-based design chart	-	37	8	3	1

the standard *np* chart is 2,071 – 36 = 2,035 when $p_0 = 0.02$, $n = 25$, ($\gamma_{\min}, \gamma_{\max}]$ = (2.0, 3.0], and MRL$_{0\min}$ = 370.4.

However, by referring to Table 3 for the out-of-control condition, the extreme percentile (the difference between the $E(\zeta_{0.05})$ and $E(\zeta_{0.95})$) reduces as n increases and the shifts interval changes from ($\gamma_{\min}, \gamma_{\max}]$ = (1.1, 2.0] (small shifts interval) to ($\gamma_{\min}, \gamma_{\max}]$ = (2.0, 3.0] (large shifts interval), for both standard *np* and DS *np* charts. This trend suggests that there is small variation for the out-of-control RL over large shift interval and larger n values. For example, the out-of-control extreme percentile of the DS *np* chart is 103.17 when MRL$_{0\min}$ = 370.4, $p_0 = 0.02$, $n = 25$, and ($\gamma_{\min}, \gamma_{\max}]$ = (1.1, 2.0], diminishes to 19.37 when ($\gamma_{\min}, \gamma_{\max}]$ = (2.0, 3.0], for identical p_0, n, and MRL$_{0\min}$. In addition, the numerical results reveal that the optimal DS *np* chart has smaller variation in RL distribution compared to the competing standard *np* chart for small and large shift interval.

Performance of the DS *np* chart when shift size is unknown

The most interesting finding emerges from the analysis shown in Tables 4–6, utilizing the optimal parameters by minimizing EMRL$_1$ to compute the MRL$_1$ when unknown shift size is a viable option, providing $\gamma \epsilon (\gamma_{\min}, \gamma_{\max}]$. The optimal charting parameters for the EMRL-based DS *np* chart can be obtained from Tables 1, 2. For ease of reference and comparison, the MRL_1 of MRL-based design chart found by Tuh et al. [22] is listed in Tables 4–6. For a comprehensive comparison, the MRL_1 values for both MRL-based and EMRL-based design charts when $\gamma = 1.2$ are also added to this section. Due to space constraint, we only present the results with $n = 100$.

From Tables 4–6, it is worth noting that the MRL_1 computed in Tables 4–6 by means of $(n_1, n_2, \mathrm{WL}, \mathrm{CL}_1, \mathrm{CL}_2)$ for DS *np* chart with EMRL-based design is nearly identical to those based on specific shift sizes (MRL-based design chart) for most cases, on condition that $\gamma \in (\gamma_{\min}, \ \gamma_{\max}]$. For instance, in Table 5, when $n = 100$, $\mathrm{MRL}_{0\min} = 200$, $p_0 = 0.02$, and $(\gamma_{\min}, \gamma_{\max}] = (1.1, \ 2.0]$, the optimal charting parameters of the DS *np* chart are $(n_1, \ n_2, \ \mathrm{WL}, \ \mathrm{CL}_1, \ \mathrm{CL}_2) = (39, \ 1{,}427, \ 2.5, \ 5.5, \ 39.5)$ (see Table 1), obtained by minimizing EMRL_1. This optimal charting parameters yield $\mathrm{MRL}_1 = \{29, 7, 4\}$ for $\gamma = \{1.2, \ 1.5, \ 2.0\}$, while the MRL-based design chart gives $\mathrm{MRL}_1 = \{28, 7, 2\}$. As a result, the optimal parameters listed in Tables 1, 2 (as determined by minimizing EMRL_1) can be directly and reliably substituted for the optimal parameters by assuming a known shift size, in the event that $\gamma \in (\gamma_{\min}, \ \gamma_{\max}]$.

An illustrative example

The performance of the DS *np* chart is assessed with the use of an example, as follows. The information used in this illustration was extracted from Gitlow and Hertz [42]. The information is relevant to the keypunching operation that normally takes place in a data processing department. To establish the control chart, a sample size of 200 cards ($n = 200$) was selected at random from the output of each day's production over the course of 24 days (subgroups $m = 24$) and inspected for defects. After establishing the control chart, it was discovered that samples 8 and 22 were not within the control limits and were subsequently discarded following further investigation. Using the remaining samples of $m = 22$ and $n = 200$, revised control limits were computed. All the verified points fall within the control limits, pointing toward in-control process. This represents phase I analysis. As a result, we may estimate the in-control process fraction nonconforming (p_0) using following equation:

$$p_0 = \frac{\sum_{i=1}^{m} p_i}{m} = \frac{\sum_{i=1}^{m} d_i}{m \times n} = \frac{73}{22 \times 200} \approx 0.02 \qquad (17)$$

We illustrate the proposed optimal EMRL-based DS *np* chart by applying a simulated data generated using the RStudio software. Herein, we use the optimal charting parameters based on $\mathrm{MRL}_{0\min} = 200$, $(\gamma_{\min}, \gamma_{\max}] = (1.1, \quad 2.0]$, $p_0 = 0.02$, and $n = 200$ obtained from Table 1. The optimal parameter combination for the DS *np* chart is $(n_1, \ n_2, \ \mathrm{WL}, \ \mathrm{CL}_1, \ \mathrm{CL}_2) = (101, \ 1{,}882, \ 4.5, \ 9.5, \ 52.5)$. The data for the 30 samples are simulated, where the first eight samples come from the in-control state with $p_0 = 0.02$. The subsequent 22 samples depict the out-of-control state with $p_1 = \gamma p_0 = 1.3 \times 0.02 = 0.026$, where a process shift of $\gamma = 1.3$ is presumed to have occurred. Note that the number of nonconforming items in the first sample d_1 is simulated from the binomial distribution with parameters $(n_1, \ p_0) = (101, \ 0.02)$ and $(n_1, \ p_1) = (101, \ 0.026)$ for the in-control and out-of-control states, respectively, while the number of nonconforming items for the second sample d_2 is generated from the same distribution but with parameters $(n_2, \ p_0) = (1{,}882, \ 0.02)$ and $(n_2, \ p_1) = (1{,}882, \ 0.026)$ for the in-control and out-of-control cases, respectively.

TABLE 7 Dataset for the illustrative example.

Sample number	DS *np* chart		
	d_1	d_2	$d_1 + d_2$
1	2		
2	0		
3	2		
4	1		
5	2		
6	1		
7	5	36	41
8	3		
9	2		
10	1		
11	3		
12	1		
13	2		
14	1		
15	6	54	60
16	4		
17	2		
18	4		
19	1		
20	2		
21	3		
22	1		
23	2		
24	1		
25	3		
26	0		
27	1		
28	0		
29	1		
30	7	40	47

The thirty samples from Table 7 are plotted in Figure 2's DS *np* chart. The solid dots (●) and hollow dots (○) represent the stages 1 and 2 of the DS scheme, respectively. One can observe

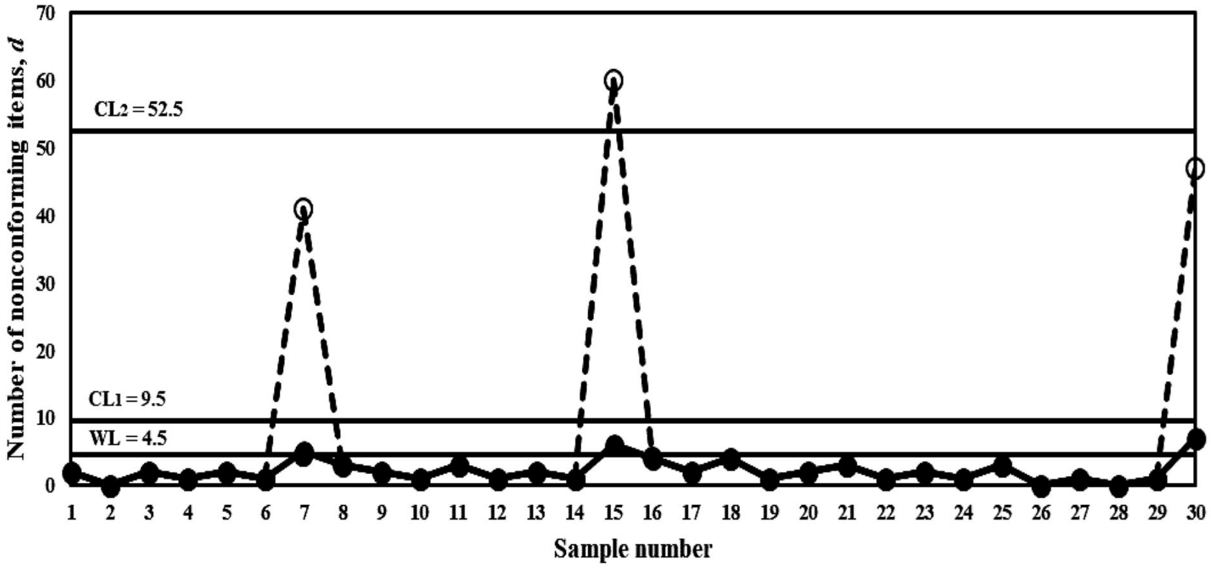


FIGURE 2
DS *np* chart.

that the process remains at the stage 1 of the DS scheme for samples 1 through 6 as the points lie lower than 4.5 (<WL) and is deemed to be in-control. Note that at sample 7, $d_1 = 5$ for the first sample at the stage 1 of the DS scheme corresponds to size $n_1 = 101$. Since $4.5 < d_1 < 9.5$, the operation moves to the stage 2 of the DS scheme, which involves taking a second sample of size $n_2 = 1{,}882$ and number of nonconforming items $d_2 = 36$ is observed. As a result, $d_1 + d_2 = 5 + 36 = 41$. Since $d_1 + d_2$ is below 52.5 ($<\text{CL}_2$), this sample is considered as in-control. The process remains in-control condition up to sample 14. At sample 15, $d_1 = 6$, $d_2 = 54$ in which $d_1 + d_2 = 60$ exceeds the control limit CL_2 of 52.5. This indicates that sample 15 is out-of-control. Clearly, DS *np* chart detects the process shift at sample 15. Corrective action should be taken immediately to identify and remove the assignable cause(s) that resulting to the out-of-control condition in the process.

Conclusion

A good understanding of a control chart is crucial as it helps to increase the confidence of quality practitioners. Therefore, in this study, EMRL has been proposed as a performance measure for designing DS *np* chart. The results obtained indicate that the EMRL is an effective optional performance measure for the DS *np* chart when it is not possible to specify the shift size of the fraction nonconforming beforehand. Alternatively, practitioners can utilize the recommended optimal charting parameters based on EMRL_1 minimization if the process shift size is within the acceptable range ($\gamma_{\min}, \gamma_{\max}]$. In the case of inexperienced practitioners who are not familiar with the establishment of process shift size, this approach can help to minimize inaccuracy that may arise when practicing and implementing the DS *np* control chart. It should be noted that the conclusion in this research depends on the data independence and binomially distributed assumptions. For future research purposes, additional work can be carried out without applying these assumptions. In addition, the effect of parameter estimation may also be conducted for the unknown shift size.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

Author contributions

MT: conceptualization, methodology, software, formal analysis, investigation, data curation, project administration, and writing—original draft. CK and HC: software, validation, resources, and writing—reviewing and editing. ML: validation, visualization, and writing—reviewing and editing. All authors contributed to the article and approved the submitted version.

Acknowledgments

We gratefully acknowledge Dr. Robin Chang Yee Hui for lending us his personal computing resources to carry out part of the calculations shown in this study.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Montgomery DC. *Introduction to Statistical Quality Control*. New York: John Wiley & Sons (2020).

2. Kumar PS. A simple method for solving type-2 and type-4 fuzzy transportation problems. *Int J Fuzzy Logic Intell Syst.* (2016) 16:225–37. doi: 10.5391/IJFIS.2016.16.4.225

3. Kumar PS. Algorithms for solving the optimization problems using fuzzy and intuitionistic fuzzy set. *Int J Syst Assur Eng Manage.* (2020) 11:189–222. doi: 10.1007/s13198-019-00941-3

4. Kumar PS. Intuitionistic fuzzy solid assignment problems: a software-based approach. *Int J Syst Assur Eng Manage.* (2019) 10:661–75. doi: 10.1007/s13198-019-00794-w

5. Kumar PS, Hussain RJ. Computationally simple approach for solving fully intuitionistic fuzzy real life transportation problems. *Int J Syst Assur Eng Manage.* (2016) 7:90–101. doi: 10.1007/s13198-014-0334-2

6. Kumar PS. Computationally simple and efficient method for solving real-life mixed intuitionistic fuzzy solid assignment problems. *Int J Fuzzy Syst Appl*. (2022)

7. Shah S, Shridhar P, Gohil D. Control chart: a statistical process control tool in pharmacy. *Asian J. Pharm.* (2014) 4:184–92. doi: 10.4103/0973-8398.72116

8. Gan FF. An optimal design of cusum control charts for binomial counts. *J Appl Stat.* (1993) 20:445–60. doi: 10.1080/02664769300000045

9. Gan FF. Monitoring observations generated from a binomial distribution using modified exponentially weighted moving average control chart. *J Stat Comput Simul.* (1990) 37:45–60. doi: 10.1080/00949659008811293

10. Epprecht EK, Costa AFB. Adaptive sample size control charts for attributes. *Qual Eng.* (2001) 13:465–73. doi: 10.1080/08982110108918675

11. Luo H, Wu Z. Optimal np control charts with variable sample sizes or variable sampling intervals. *Econ. Q. Cont.* (2002) 17:39–61. doi: 10.1515/EQC.2002.39

12. Croasdale R. Control charts for a double-sampling scheme based on average production run lengths. *Int. J. Prod. Res.* (1974) 12:585–92. doi: 10.1080/00207547408919577

13. Daudin JJ. Double sampling X charts. *J Qual Technol.* (1992) 24:78–87. doi: 10.1080/00224065.1992.12015231

14. He D, Grigoryan A. An improved double sampling s chart. *Int J Prod Res.* (2003) 41:2663–79. doi: 10.1080/0020754031000093187

15. Costa AFB, Claro FAE. Double sampling X control chart for a first-order autoregressive moving average process model. *Int J Adv Manuf Technol.* (2008) 39:521–42. doi: 10.1007/s00170-007-1230-6

16. Torng CC, Lee PH. The performance of double sampling control charts under non-normality. *Commun Stat Simul Comput.* (2009) 38:541–57. doi: 10.1080/03610910802571188

17. Khoo MBC, Lee HC, Wu Z, Chen CH, Castagliola P. A synthetic double sampling control chart for the process mean. *IIE Transac.* (2010) 43:23–38. doi: 10.1080/0740817X.2010.491503

18. De Araujo Rodrigues AA, Epprecht EK, De Magalhaes MS. Double sampling control charts for attributes. *J Appl Stat.* (2011) 38:87–112. doi: 10.1080/02664760903266007

19. Chong ZL, Khoo MBC, Castagliola P. Synthetic double sampling np control chart for attributes. *Comput Indus Eng.* (2014) 75:157–69. doi: 10.1016/j.cie.2014.06.016

20. Joekes S, Smrekar M, Barbosa EP. Extending a double sampling control chart for non-conforming proportion in high quality processes to the case of small samples. *Stat Methodol.* (2015) 23:35–49. doi: 10.1016/j.stamet.2014.09.003

21. Lee MH, Khoo MBC. Double sampling np chart with estimated process parameter. *Commun Stat Simul Comput.* (2021) 50:2232–50. doi: 10.1080/03610918.2019.1599017

22. Tuh MH, Lee MH, Lau EMF, Then PHH. Performance of the double sampling *np* chart based on the median run length. *Adv Math Sci J.* (2020) 9:7429–38. doi: 10.37418/amsj.9.9.89

23. Teoh WL, Khoo MBC, Castagliola P, Chakraborti S. Optimal design of the double sampling X chart with estimated parameters based on median run length. *Comput Indus Eng.* (2014) 67:104–15. doi: 10.1371/journal.pone.0068580

24. Khoo MBC, Wong VH, Wu Z, Castagliola P. Optimal design of the synthetic chart for the process mean based on median run length. *IIE Transac.* (2012) 44:765–79. doi: 10.1080/0740817X.2011.609526

25. Lee MH, Khoo MBC. Optimal designs of multivariate synthetic |S| control chart based on median run length. *Commun Stat Theor Methods.* (2017) 46:3034–53. doi: 10.1080/03610926.2015.1048884

26. Smajdorová T, Noskievičová D. Analysis and application of selected control charts suitable for smart manufacturing processes. *Appl Sci.* (2022) 12:5410. doi: 10.3390/app12115410

27. Graham MA, Chakraborti S, Mukherjee A. Design and implementation of cusum exceedance control charts for unknown location. *Int J Prod Res.* (2014) 52:5546–64. doi: 10.1080/00207543.2014.917214

28. Chakraborti S. Run length distribution and percentiles: the Shewhart chart with unknown parameters. *Qual Eng.* (2007) 19:119–27. doi: 10.1080/08982110701276653

29. Lee MH, Khoo MBC. Optimal design of synthetic np control chart based on median run length. *Commun Stat Theor Methods*. (2017) 46:8544–56. doi: 10.1080/03610926.2016.1183790

30. Faraz A, Saniga E, Montgomery D. Percentile-based control chart design with an application to Shewhart X and S^2 control charts. *Qual Reliab Eng Int.* (2019) 35:116–26. doi: 10.1002/qre.2384

31. Gao H, Khoo MBC, Teh SY, Teoh WL. A study on the median run length performance of the run sum s control chart. *Int J Mech Eng Robot Res.* (2019) 8:885–90. doi: 10.18178/ijmerr.8.6.885-890

32. Qiao Y, Hu X, Sun J, Xu Q. Optimal design of one-sided exponential ewma charts with estimated parameters based on the median run length. *IEEE Access.* (2019) 7:76645–58. doi: 10.1109/ACCESS.2019.29 21427

33. You HW, Khoo MBC, Castagliola P, Qu L. Optimal exponentially weighted moving average charts with estimated parameters based on median run length and expected median run length. *Int J Prod Res.* (2016) 54:5073–94. doi: 10.1080/00207543.2016.1145820

34. Teoh WL, Chong JK, Khoo MBC, Castagliola P, Yeong WC. Optimal designs of the variable sample size chart based on median run length and expected median run length. *Qual Reliab Eng Int.* (2017) 33:121–34. doi: 10.1002/qre.1994

35. Tang A, Castagliola P, Sun J, Hu X. Optimal design of the adaptive ewma chart for the mean based on median run length and expected median run length. *Qual Technol Quant Manag.* (2019) 16:439–58. doi: 10.1080/16843703.2018.1460908

36. Chong ZL, Tan KL, Khoo MBC, Teoh WL, Castagliola P. Optimal designs of the exponentially weighted moving average (ewma) median chart for known and estimated parameters based on median run length. *Commun Stat Simul Comput.* (2022) 51:3660–84. doi: 10.1080/03610918.2020.1721539

37. Yeong WC, Lee PY, Lim SL, Ng PS, Khaw KW. Optimal designs of the side sensitive synthetic chart for the coefficient of variation based on the median run length and expected median run length. *PLoS ONE.* (2021) 16:e0255366. doi: 10.1371/journal.pone.0255366

38. Brook D, Evans DA. An approach to the probability distribution of cusum run length. *Biometrika.* (1972) 59:539–49. doi: 10.1093/biomet/59.3.539

39. Castagliola P, Celano G, Psarakis S. Monitoring the coefficient of variation using ewma charts. *J Qual Technol.* (2011) 43:249–65. doi: 10.1080/00224065.2011.11917861

40. Winckel G. Legendre-Gauss Quadrature Weights and Nodes. *MATLAB Central File Exchange.* Available online at: https://www.mathworks.com/matlabcentral/fileexchange/4540-legendre-gauss-quadrature-weights-and-nodes (accessed on January 5, 2022).

41. Hale N, Townsend A. Fast and accurate computation of Gauss-Legendre and Gauss-Jacobi Quadrature nodes and weights. *SIAM J Sci Comput.* (2013) 35:A652–74. doi: 10.1137/120889873

42. Gitlow HS, Hertz PT. *Product Defects and Productivity.* (1983). Available online at: https://hbr.org/1983/09/product-defects-and-productivity (accessed on June 22, 2022).

TYPE Original Research
PUBLISHED 20 October 2022
DOI 10.3389/fams.2022.955044

Predicting the data structure prior to extreme events from passive observables using echo state network

Abhirup Banerjee[1,2]*, Arindam Mishra[3,4], Syamal K. Dana[3,5], Chittaranjan Hens[6,7], Tomasz Kapitaniak[3], Jürgen Kurths[1,3,8] and Norbert Marwan[1,9]

[1]Complexity Science, Potsdam Institute for Climate Impact Research, Potsdam, Germany, [2]Institute for Physics and Astronomy, University of Potsdam, Potsdam, Germany, [3]Division of Dynamics, Lodz University of Technology, Łódź, Poland, [4]Department of Physics, National University of Singapore, Singapore, Singapore, [5]Department of Mathematics, National Institute of Technology, Durgapur, India, [6]Physics and Applied Mathematics Unit, Indian Statistical Institute, Kolkata, India, [7]Center for Computational Natural Sciences and Bioinformatics, International Institute of Information Technology Gachibowli, Hyderabad, India, [8]Institute of Physics, Humboldt Universität zu Berlin, Berlin, Germany, [9]Institute of Geoscience, University of Potsdam, Potsdam, Germany

Check for updates

OPEN ACCESS

EDITED BY
Ulrich Parlitz,
Max Planck Society, Germany

REVIEWED BY
Dibakar Ghosh,
Indian Statistical Institute, India
Nahal Sharafi,
Hamburg University of Applied Sciences, Germany

*CORRESPONDENCE
Abhirup Banerjee
banerjee@pik-potsdam.de

SPECIALTY SECTION
This article was submitted to Dynamical Systems, a section of the journal Frontiers in Applied Mathematics and Statistics

RECEIVED 28 May 2022
ACCEPTED 30 September 2022
PUBLISHED 20 October 2022

CITATION
Banerjee A, Mishra A, Dana SK, Hens C, Kapitaniak T, Kurths J and Marwan N (2022) Predicting the data structure prior to extreme events from passive observables using echo state network. *Front. Appl. Math. Stat.* 8:955044. doi: 10.3389/fams.2022.955044

Extreme events are defined as events that largely deviate from the nominal state of the system as observed in a time series. Due to the rarity and uncertainty of their occurrence, predicting extreme events has been challenging. In real life, some variables (passive variables) often encode significant information about the occurrence of extreme events manifested in another variable (active variable). For example, observables such as temperature, pressure, etc., act as passive variables in case of extreme precipitation events. These passive variables do not show any large excursion from the nominal condition yet carry the fingerprint of the extreme events. In this study, we propose a reservoir computation-based framework that can predict the preceding structure or pattern in the time evolution of the active variable that leads to an extreme event using information from the passive variable. An appropriate threshold height of events is a prerequisite for detecting extreme events and improving the skill of their prediction. We demonstrate that the magnitude of extreme events and the appearance of a coherent pattern before the arrival of the extreme event in a time series affect the prediction skill. Quantitatively, we confirm this using a metric describing the mean phase difference between the input time signals, which decreases when the magnitude of the extreme event is relatively higher, thereby increasing the predictability skill.

KEYWORDS

extreme events, coupled neuron model, active and passive variable, precursory structure, echo state network, phase coherence

Introduction

In recent years, extreme events (EEs) have gained attention of the researchers and decision-makers due to increase in the occurrence of highly intense climate extremes, such as hurricanes, floods, heatwaves, etc., due to global warming and climate change [1–3]. They have devastating impact on life and infrastructure. There are several other examples of such extraordinary devastating events in various other disciplines aside from climate, like rogue waves in lasers and tsunamis in the ocean, earthquakes in seismology, share market crashes in finance, regime shift in ecosystems, etc., which are also rare but may have a long-term correlation in their return periods [4–11]. The study of extreme events focuses on the self-organizing principles [5, 12–19] that may enable us to forecast and mitigate the after effect. Various tools have been developed to study the underlying dynamics of such extreme events, e.g., complex networks have been extensively used to analyze climate extremes [20–24], numerous studies have been conducted to analyze extreme events based on their statistical properties [25, 26]. Recurrence plot analysis has been used to study the recurring behavior of flood events [27]. Because of their rare occurrence and complex dynamics, understanding and predicting extreme events is a challenge in the studies of complex natural systems using the dynamical system approach only [15, 28–30]. Alternatively, data-based and model-free machine learning techniques have been recently shown to be more promising for predicting such events [31–36]. To put it simply, such a prediction process involves training of the machine using past data records of EEs from other observable and then testing the ability of machine to successfully predict the prior shape of the observable which leads to extreme event.

As the term "extreme event" is used in various disciplines, a precise definition of EEs is not available. Rather, it depends on the particular discipline where this term is being used. In this work, we select the EEs based on their magnitude. Therefore, it is crucial to set a threshold height so that we can call an event "extreme" when it exceeds the threshold. The choice of an appropriate threshold plays a pivotal role in prediction [37, 38]. In our study, we found that for data-based machine learning, a certain threshold height augments the efficient detection of the arrival of a coherent pattern and thereby leverage the prediction process. In particular, we raise the following question here that for a given multivariate data set in which one of the variables exhibits EEs, whether a seemingly benign variable (with no signature of EE) can be used in a machine for the prediction of the preceding structure or pattern indicative of the forthcoming EE expressed in another observed variable. We refer to the preceding structure pattern as a precursory signal in the data that is typically correlated with the occurrence of EE in near future. For example, farmers anticipate rain when they observe red clouds in the early morning sky.

The aforementioned question is motivated from the fact that the occurrence of EEs in one variable are a manifestation of the rich dynamics of a multivariate higher dimensional system as caused by the non-linear interactions among its various constituents [5, 13]. Due to the paucity of observations of some EEs occurring in nature, collection or reconstruction of data directly from a dynamic variable that flares up with an extreme value (*active* variable) such as the extreme precipitation, over a long time period is seldom possible. It is easier to reconstruct data for those observables which are slow varying (temperature, pressure, etc.). Some of these observables may remain silent or *passive* with a weak response and do not show up with any manifestation of large size extreme value. However, such *passive* variables carry significant information related to the EEs. We emphasize here that the data collected from the passive variable is used as inputs to a reservoir computing machine, i.e., the echo-state network (ESN), in order to check how efficiently the machine can capture the a priori structure in the *active* observable that precedes the EE. ESN is a simple version of recurrent neural networks [39] that has been used extensively to predict complex signals ranging from time series generated from chaotic model, stock-price data to tune hyperparameter [40–50]. Recently, it has been shown that ESN can efficiently capture the onset of generalized synchronization [51–55], quenching of oscillation [56, 57], detect collective bursting in neuron populations [58], and predict epidemic spreading [59]. ESN has been shown to have great potential in handling multiple inputs of temporal data, and ability to trace the relation between them [52, 58, 60]. Due to its simple and computationally effective character and its suitability for dynamical systems, we use ESN for our study. Other machine learning-based methods, such as *deep learning* [61] might also be useful for the problem we address in this work.

While collecting data, the first important task is to detect the EE by assigning an appropriate threshold height and collect a number of data segments prior to all the available EE in a time series, to address the question of predictability as suggested earlier [38, 62, 63]. In the present work, we rely on data generated from numerical simulations of a model system for training and testing of the ESN for efficient detection of the structure preceding the extreme events. Firstly, we identify a large number of visible EEs from the active variable using a threshold height and save a data segment of identical length prior to the occurrence of each EE from the active variable along with the corresponding data segment from the passive variable. A multiple number of data segments of identical length corresponding to EEs in the active variable are thus collected from the passive variable and used as inputs to the machine. A part or fraction of the data points from each segment is used for training and the rest of the data points is kept aside for predicting the preceding structure of EE in the active variable during testing.

We repeat the whole process of data collection, training and testing of the machine by varying the choice of the threshold height and then make a quantitative comparison based on predictability skill to select the most suitable threshold height for detection and prediction of EE. It must be noted that by prediction we imply the identification of a common pattern or structure in the test signal that always appears quite ahead of time before the arrival of extreme events and hence, effectively works as a precursor to the extreme events. Our machine learning based recipe unfolds two useful information: (i) Data collected from a passive variable before the appearance of EE in an active variable can provide clues to capture the future trend of an active variable and thereby predict the precursory shape of the forthcoming EE, (ii) machine can efficiently suggest a choice of appropriate threshold height that may augment the prediction process. A possible reason for the necessity of a critical threshold for accurate prediction by the machine is explained further in light of a coherent pattern that always appears in the ensemble of multiple segments of data inputs that has been collected prior to the EE.

For demonstration purpose, we use a paradigmatic model neuron that consists of active variables (fast variables) expressing the triggering of extreme events when its passive counterpart (slow variable) shows no signature of extremes.

Methodology

Dataset

For data generation of EEs, we numerically simulate a synaptically (chemically) coupled slow-fast Hindmarsh-Rose (HR) neurons model [64],

$$\begin{aligned}\dot{x_i} &= y_i + bx_i^2 - ax_i^3 - z_i + I - \theta_i(x_i - v_s)\Gamma(x_j)\\ \dot{y_i} &= c - \kappa x_i^2 - y_i \\ \dot{z_i} &= \rho[s(x_i - x_R) - z_i],\end{aligned} \tag{1}$$

where x_i and y_i $(i, j = 1, 2; i \neq j)$ are the fast variables and oscillate with firing of spiking or bursting potentials. The slow variable z_i controls the fast oscillations. Each variable has its specific biological functional meaning. The system parameters a, b, c, and s are appropriately chosen where $r < 1$ is the slow parameter. x_R and v_s are constant biases and $\Gamma(x) = \frac{1}{1+e^{-\lambda(x-\Theta)}}$ is a sigmoidal function, typically used [65] to represent chemical synaptic coupling. The parameters, $a = 1, b = 3, c = 1, \kappa = 5, x_R = -1.6, \rho = 0.01, s = 5, I = 4, v_s = 2, \lambda = 10, \Theta = -0.25$, are kept fixed for generating data. The coupling constant $\theta_{1,2}$ decides the strength of mutual communication between the neurons *via* chemical synapses. We collect data on x_i and z_i $(i = 1, 2)$ from numerical simulations and define two new variables, $u = x_1+x_2$ and $v = z_1+z_2$. Extreme events are expressed [65] in the fast variable u, which is denoted as our active variable, while the slow variable v is defined as the passive variable. The passive variable does show a signature of rising amplitude when extreme events arrive in the active variable. However, we have to make a cut-off in the range of the threshold as usually used from 4σ to 8σ in the literature. The rising peaks in the slow variable are not significantly large than our considered significant height (3.5σ to 6σ). Our motivation is to predict the precursory structures for rare peaks, and for this purpose, we consider the v variable as a passive variable. Information from the passive variable v is then used as input data to the machine for predicting the preceding structure of extremes in u.

The local maxima of a time series are identified as events and accordingly all the events are extracted from u for a long run. A standard definition is used for the identification of an extreme event [14, 15, 66] with a threshold $H_s = \langle\mu\rangle + d\sigma$, where $\langle\mu\rangle$ is the mean of the time series, σ is the standard deviation and d is a constant. Any event larger than H_s is considered as an extreme where d is allowed to vary from system to system or for a measured time series under consideration. The question of prediction and enhancing predictability is addressed here by setting different threshold limits of H_s by varying d.

For the purpose of numerical experimentation, we first detect a number of extreme peaks n from a long time series of u (total length of the time series : 2×10^7) that crosses a predefined threshold H_s for a particular choice of d. Next, we collect k data points prior to each of the n peaks from u, i.e.,

$$\begin{aligned}\hat{u}_1 &= (u_1(t), u_1(t-1), u_1(t-2), ..., u_1(t-k))\\ \hat{u}_2 &= (u_2(t), u_2(t-1), u_2(t-2), ..., u_2(t-k))\\ &:\\ &:\\ \hat{u}_n &= (u_n(t), u_n(t-1), u_n(t-2), ..., u_n(t-k)),\end{aligned} \tag{2}$$

where $\hat{u}_1, \hat{u}_2, ..., \hat{u}_n$ are the n events selected from active variable u. We also collect the corresponding data points from the v-time series, i.e.,

$$\begin{aligned}\hat{v}_1 &= (v_1(t), v_1(t-1), v_1(t-2), ..., v_1(t-k))\\ \hat{v}_2 &= (v_2(t), v_2(t-1), v_2(t-2), ..., v_2(t-k))\\ &:\\ &:\\ \hat{v}_n &= (v_n(t), v_n(t-1), v_n(t-2), ..., v_n(t-k)).\end{aligned} \tag{3}$$

In other words, we collect n time segments each containing k data points prior to all the n extreme events, and construct a matrix called event matrix E of size $n \times k$ from the active variable and, similarly, construct a matrix P of the same size $n \times k$ by storing the corresponding data points from the passive variable. A set of m $(m < n)$ (gray region A in Figure 2A) time segments each with data points p $(p < k)$ (Figure 2B) as collected from v is then fed into the machine for training to predict the preceding structure of $(n - m)$ segments in u signals (light red region B in Figure 2), which is considered as a precursor to the arrival of extreme events later. How the machine extracts information from the inputs of v and transforms them into u at the output is defined in the input-output functional relation of the machine as a description of the ESN in the next section. Once the training is over, the rest of the $(k - p)$ data points for each of the m time segments are used for testing whether the machine can predict the future structure of $(n - m)$ time segments of u. The whole process is repeated multiple times by using four different choices of d (3.5, 4, 5, 6) for detecting extremes from the time series of u. We emphasize once again that an input to the machine for training and testing consists of multiple segments of data points of identical length collected from v corresponding to the successive number of EE detected in u for each d-value. The data points collected from u are used at a later stage for comparison with the machine output during the testing process. Certainly this recipe works only when certain amount of data prior to the extreme events is available from both the variables, and the passive variable of the system can be identified. However, the advantage of such a methodology is that it is data-driven and model-free.

Reservoir computing: Echo-state network model

An echo state network (ESN) is a type of recurrent neural network and is extensively used due to its simple architecture [39]. It has three parts—(1) input layer—in which the weights are randomly chosen and fixed, (2) reservoir or hidden layer—it is formed by randomly and sparsely connected neurons and (3) output layer—in which the output weights are the only trainable part by input data. A standard leaky network with a tanh activation function is considered here as the ESN. The dynamics of each reservoir node is governed by the following recursive relation:

$$\mathbf{r}(t+1) = (1-\alpha)\mathbf{r}(\mathrm{t}) + \alpha \tanh\left(\mathbf{W}_{\mathrm{res}}\mathbf{r}(\mathrm{t}) + \mathbf{W}_{\mathrm{in}}[1; \mathbf{v}(\mathrm{t})]\right), \quad (4)$$

where $\mathbf{r}(t)$ is a n_{res}-dimensional vector that denotes the state of the reservoir nodes at time instant t, $\mathbf{v}(t)$ is the m-dimensional input vector and 1 is the bias term. The matrices $\mathbf{W}_{\mathrm{res}}$ $(n_{\mathrm{res}} \times n_{\mathrm{res}})$ and $\mathbf{W}_{\mathrm{in}}$ $(n_{\mathrm{res}} \times (m+1))$ represent the weights of the internal connection of the reservoir nodes and weights of the input, respectively. The parameter α is the leakage constant, which can take any values between 0 to 1. It is to be noted that the tanh function operates element-wise. The choices of α and n_{res} can be varied. Here, we have fixed $\alpha = 0.6$ and $n_{\mathrm{res}} = 600$ throughout all simulations. The reservoir weight matrix $\mathbf{W}_{\mathrm{res}}$ is constructed by drawing random numbers uniformly over an interval $[-1, 1]$ and the spectral radius of the matrix $\mathbf{W}_{\mathrm{res}}$ is re-scaled to less than unity. The elements of the input weight matrix $\mathbf{W}_{\mathrm{in}}$ are also generated randomly from the interval $[-1, 1]$. Next we consider data of n-segments sequentially from the time series of v corresponding to n extreme peaks in u from which a set of first m-segments of length p of the total length of k data points are fed into the ESN for training. Thereafter, the output weight W_{out} is optimized to capture the trend of the $(n - m)$ segments (each length: $(k - p)$) of u signals. Once the machine is trained, the input of m-segments each with $(k-p)$ data points are fed into the machine to predict the trend of the $(n - m)$-segments of the u signals prior to the arrival of EE in time. At each instant of time t, the $m-$dimensional input vector of data, $\mathbf{v}(t)$:$[v_1(t), v_2(t), ..., v_m(t)]^T$ is fed into m-number of input nodes of the machine when the contribution of the input weight matrix in the dynamics of the reservoir (see Equation 4) is written as,

$$\begin{bmatrix} \mathbf{W}_{\mathrm{in}}(1,1) & \cdots & \mathbf{W}_{\mathrm{in}}(1,m+1) \\ \mathbf{W}_{\mathrm{in}}(2,1) & \cdots & \mathbf{W}_{\mathrm{in}}(2,m+1) \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{W}_{\mathrm{in}}(n_{\mathrm{res}},1) & \cdots & \mathbf{W}_{\mathrm{in}}(n_{\mathrm{res}},m+1) \end{bmatrix} \times \begin{bmatrix} 1 \\ v_1(t) \\ v_2(t) \\ \vdots \\ v_m(t) \end{bmatrix}.$$

During the training process, at each time instant t, the reservoir state $\mathbf{r}(\mathrm{t})$ and input $\mathbf{v}(\mathrm{t})$ are accumulated in $\mathbf{V}_{\mathrm{train}}(\mathrm{t}) = [1; \mathbf{v}(\mathrm{t}); \mathbf{r}(\mathrm{t})]$. The matrix $\mathbf{V}_{train}$ having dimension $(n_{\mathrm{res}} + m + 1) \times p$ look like,

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ v(1,1) & v(1,2) & \cdots & v(1,p) \\ v(2,1) & v(2,2) & \cdots & v(2,p) \\ \vdots & \vdots & \vdots & \vdots \\ v(m,1) & v(m,2) & \cdots & v(m,p) \\ r(1,1) & r(1,2) & \cdots & r(1,p) \\ r(2,1) & r(2,2) & \cdots & r(2,p) \\ \vdots & \vdots & \vdots & \vdots \\ r(n_{\mathrm{res}},1) & r(n_{\mathrm{res}},2) & \cdots & r(n_{\mathrm{res}},p) \end{bmatrix}.$$

The output weight is determined by:

$$\mathbf{W}_{\mathrm{out}} = \mathbf{U}_{\mathrm{train}}\mathbf{V}_{\mathrm{train}}^{\mathrm{T}}(\mathbf{V}_{\mathrm{train}}\mathbf{V}_{\mathrm{train}}^{\mathrm{T}} + \lambda \mathrm{I})^{-1}, \quad (5)$$

where $\mathbf{U}_{train}$ is a matrix which stores the value of u from $(n - m)$ segments of training length p, and $\lambda = 10^{-8}$ is the regularization factor that avoids over-fitting. Now, the output weight is optimized, the final output is obtained by,

$$\mathbf{U} = \mathbf{W}_{\mathrm{out}}\mathbf{V}, \quad (6)$$

An important point to note is that we use the information of u only to optimize the output weight.

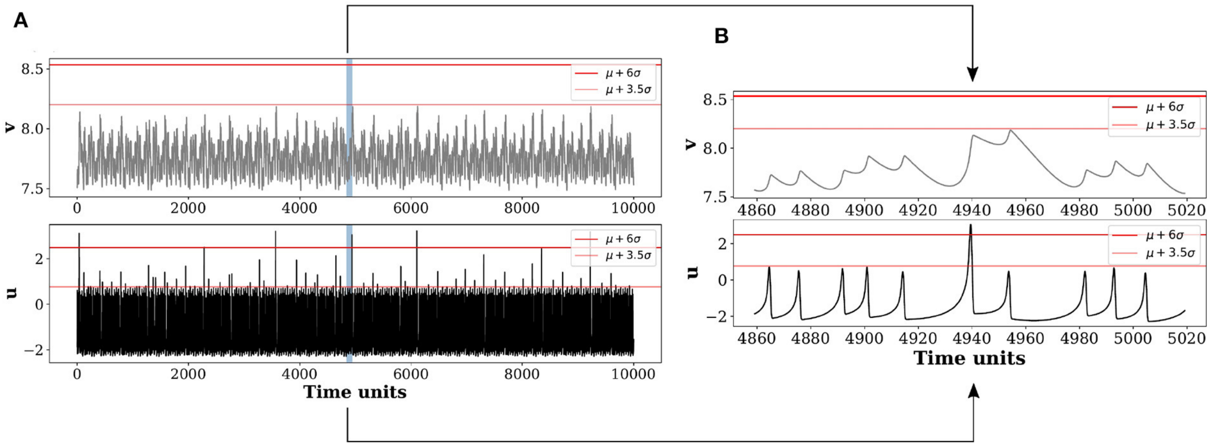


FIGURE 1
Time series of slow variable v and fast variable u of the coupled Hindmarsh-Rose (HR) system. **(A)** Horizontal red lines in the time series of *u* (lower panel) and *v* (upper panel), indicate two threshold heights $H_{s_1} = \langle\mu\rangle + 3.5\sigma$ (thin line), $H_{s_2} = \langle\mu\rangle + 6\sigma$ (bold line); μ and σ are the mean and standard deviation of the time series, respectively. Threshold height H_{s_2} filters out many large peaks that are otherwise qualified as extremes by the lower threshold H_{s_1}, and thereby allows a selection of rarer extreme events only. One particular extreme peak (shaded region) is marked in **(A)** as shown in *u*, and zoomed in the lower panel of **(B)** for illustration. This extreme peak is larger than both the horizontal lines H_{s_1} and H_{s_2} so as to qualify as a rare extreme event. The corresponding part of the time series of the slow variable *v* in the upper panel of **(A)** that never crosses either of the thresholds, H_{s_1} and H_{s_2}, is zoomed in and shown in the upper panel of **(B)**. Although a slight increase in size of the peak is seen **(B)** compared to its neighboring peaks (upper panel), there is not much significant change in height in comparison to the extreme peak observed in *u* in the lower panel.

Results

For illustration of our proposed scheme, the original time series of u and v for a long run of numerical simulations are plotted in Figure 1A. As the threshold height is increased from $H_{s_1} = \langle\mu\rangle + 3.5\sigma$ and $H_{s_2} = \langle\mu\rangle + 6\sigma$ by varying d from 3.5 to 6, many large peaks are filtered out that declares only a few peaks as rare and extremes. The extreme peaks are selected as those which are higher than a selected threshold height H_s (horizontal line, Figure 1A) for a particular choice of d, and used as data for training and testing the reservoir shown in Figures 2B–D. It is clear that some of the peaks in u are higher than the designated thresholds H_{s_1} and H_{s_2} whereas the height of all the peaks in v are lower than both thresholds. A zoomed version is shown in Figure 1B to demonstrate the time evolution of u and v around a single extreme peak marked by a shaded region in Figure 1A. Extremes are only expressed in the active variable u with no similar manifestation in the passive variable v, which is considered here as the input candidate to the machine for the prediction of the a priori structure of successive EEs in u.

An exemplary predicted output of u for $(k - p) = 200$ data points (blue circles) vis-à-vis the original u signal of the same length (blue line) is plotted in Figures 3A–D for four different d-values. A visual impression provides a clear evidence that the error between the predicted signal (blue circles) and the original input signal (blue line) during $1,300$ to $1,500$ time units decreases with the increase in the value of d. For a more comprehensive understanding of the scenario, the root mean square error (RMSE) estimated for 20 predicted output signals and the original signals of u is plotted which confirms the increasing predictability with higher H_s (Figures 3E–H). To verify the robustness of the outcome, we repeat the whole process for 400 realizations drawn from 400 different initial conditions. RMSE is calculated as follows:

$$RMSE = \sqrt{\frac{1}{t_f - t_r}\sum_{t=t_r}^{t_f}(u_{\text{original}}(t) - u_{\text{machine}}(t))^2}. \quad (7)$$

where t_r and t_f are training and final time respectively and $t_f - t_r = k - p$.

To understand the reason for the machine's improved performance with higher a H_s, we compare all the 180 input signals of the passive variable (v) as well as the active variable (u) prior to the occurrence of EEs ($p = 1,300$ data points) (Figure 4). Upper row plots in Figures 4A–D represent the input signals v before the EEs for four different threshold values. As we increase the threshold H_s (by increasing d from 3.5, 4, 5, 6), signals observed to get less dispersed and tend to form a coherent bundle.

In fact, the increasing coherent pattern among the input signals is more prominent in the corresponding active variable u in the lower row of Figures 4E–H than the v variable. For the highest threshold value, the time signals are almost coherent

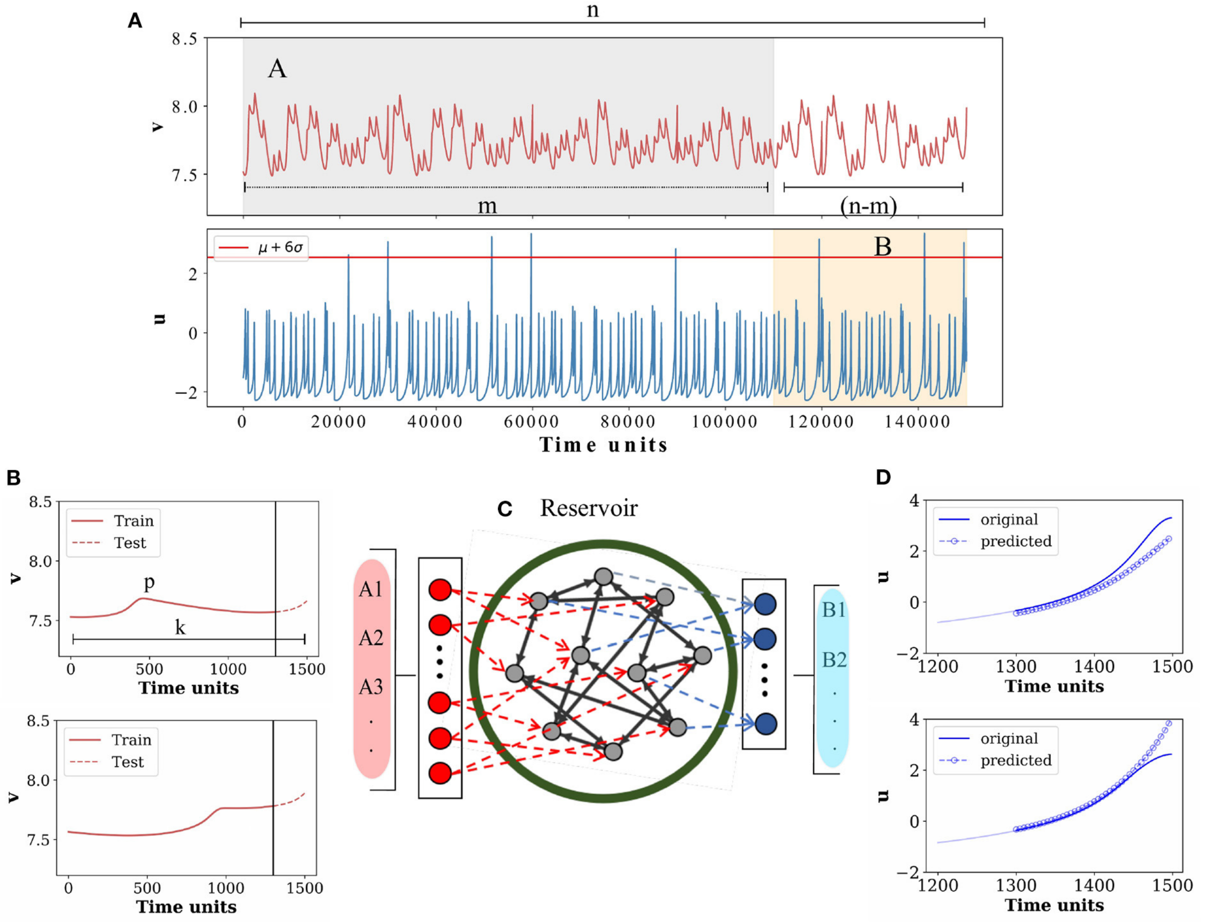


FIGURE 2
Schematic diagram of the ESN and the prediction process. **(A)** Time series of the passive variable v (upper panel) and active variable u (lower panel) with a number of extreme events, here selected using a threshold height $H_s = \mu + 6\sigma$, are shown. Data points ($k = 1,500$) from v, and u prior to n extreme peaks are saved. A few exemplary extreme peaks are shown for demonstration. For our proposed scheme, data points around such $n = 200$ extreme peaks are collected. **(B)** Two exemplary input signals corresponding to two extreme events are shown here, while the actual number of input signals are $m = 180$ as for the training purpose. For each input node, $p = 1,300$ data points (solid red line) are used for training purpose and the rest of $(k - p) = 200$ data points (dotted red line) are used for testing, which are separated by a vertical line (black line). **(C)** Echo state network structure: input layer consists of A_m nodes, where $m = 180$ input signals (data segments prior to each of the extreme events) are used for training. The output layer consists of $B_{n-m} = 20$ nodes. **(D)** Preceding pattern of predicted u signals from 20 nodes each for $(k - p) = 200$ datapoints (blue circles) and the original u signal (blue line) for 200 datapoints are plotted for comparison. Two such output signals are shown as examples.

similar to what was reported by [62], where they showed the formation of coherent structure before the arrival of extreme events in the active variables. The increasing coherence in v with higher H_s enhances the machine's predictability skill for higher amplitude events compared to the lower amplitude ones. Thus, the machine establishes a general fact, in quantitative terms, that predictability is enhanced for larger value of threshold height when the input signals are more coherent for a longer duration of time [62, 63].

We repeat our experiments using the same ESN by considering two different length of data inputs ($p = 800, 1,300$) prior to each of the extreme events for training, and keeping the same set length of data points $(k - p) = 200$ for testing as done above. The number of inputs (A_m; $m = 180$) for training and outputs for testing (B_{n-m}; $n - m = 20$) remain unchanged. Thereafter, we calculate the RMSE of the predicted output signals from 20 output nodes for each length of data inputs (p) and repeat the whole process for increasings d-values. We plot the RMSE against the d-values and for two different time lengths ($800, 1,300$) in Figure 5A. The RMSE is high for $d = 3.5$, and it gradually decreases and converges to a low value for higher threshold values. We confirm that our results machine learning framework also work for changing the number of inputs and outputs, and also by changing the length of the testing data length (see Supplementary material).

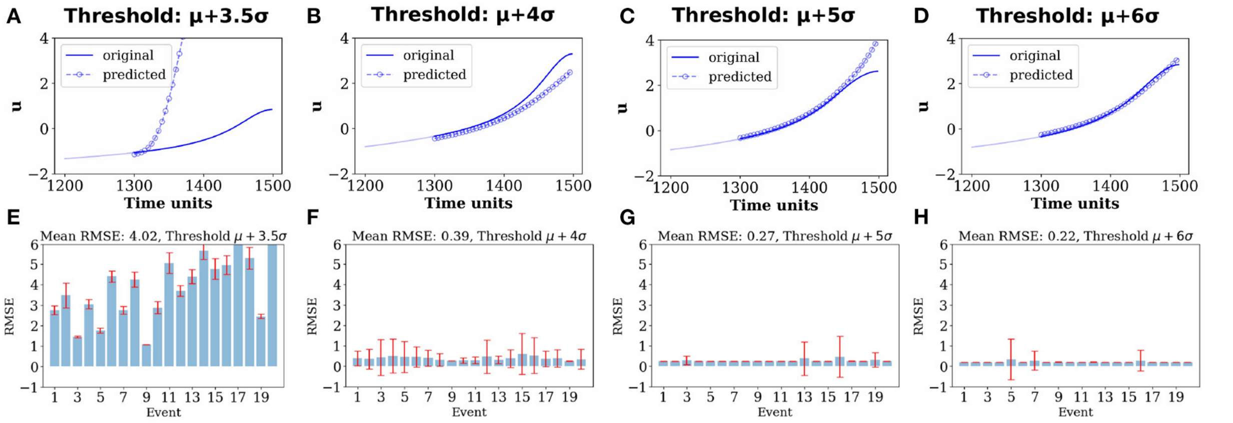


FIGURE 3
Prediction of extreme events by the ESN. Upper panels in **(A–D)** show original active signal u for $(k-p) = 200$ data points (blue line) along with the predicted signal for $(k-p) = 200$ data points (blue circles) for comparison for EEs selected using four different threshold heights computed using: **(A)** $d = 3.5$, **(B)** $d = 4$, **(C)** $d = 5$, and **(D)** $d = 6$. It shows an increased resemblance between the predicted and original extreme peaks with increasing d. Lower panels in **(E–H)** show RMSE between the original signal u and their predicted signals for $(k-p) = 200$ data during testing, estimated over 20 extreme events, corresponding to **(A–D)**, respectively. Results of 400 realizations of data from numerical simulations of the model using 400 different initial conditions for each d-value are presented in **(E–H)** and the vertical bars mark their standard deviation.

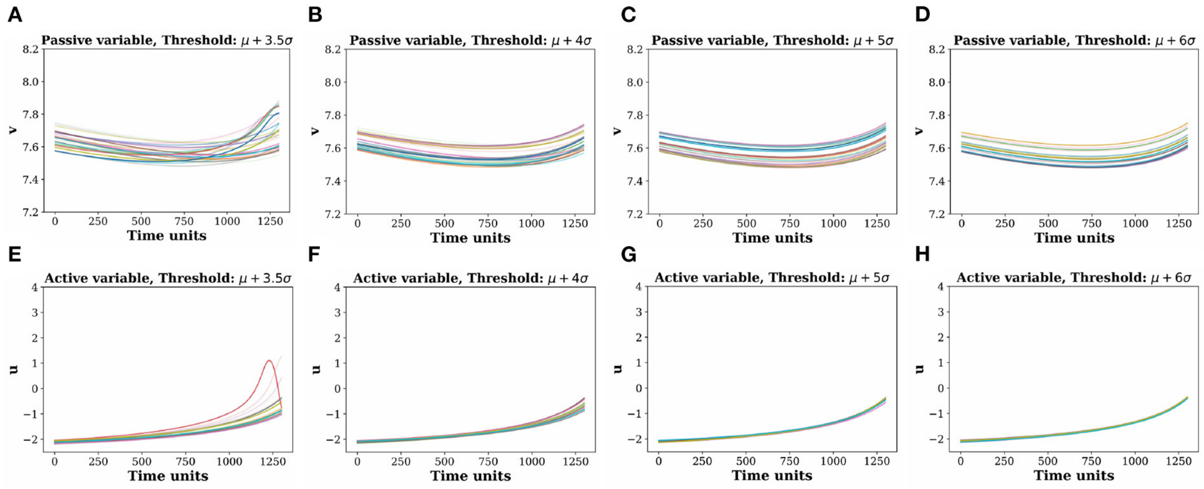


FIGURE 4
Comparative picture of coherence in the input time signals (p) extracted before an extreme events. **(A–D)** Input signal of passive variable v for threshold values ($d = 3.5, 4, 5, 6$). **(E–H)** are the corresponding active variable u for threshold values ($d = 3.5, 4, 5, 6$). Coherence between the input time signals increases with the threshold height determined by higher d-values. Different color signifies different trajectories.

Next we introduce another measure ψ based on the instantaneous phases of the time signal inputs,

$$\psi = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{\substack{j=i+1 \\ j \neq i}}^{n} \left[\frac{1}{T} \sum_{t=1}^{T} \mid \phi_i(t) - \phi_j(t) \mid \right] \tag{8}$$

where $\phi_i(t)$ is the instantaneous phase of the i-th signal of the passive variable v at time t, n is the total number of segments and T is the segment length. Here, $\phi_i(t)$ of i th signal is calculated using the Hilbert transform [67]. High value of ψ indicates less coherent structure and vice-versa. This variable ψ represents the average phase difference (on the number of segment and segment length) between all the 180 input signals of different length.

We plot values of ψ against d for the two different time lengths (800, 1, 300) in Figure 5B. A phase coherence is observed with increasing d. When the threshold is low (lower value of d), the time signals of v are dispersed (see Figure 5A). As a result, the average phase difference ψ is high. ψ gradually converges

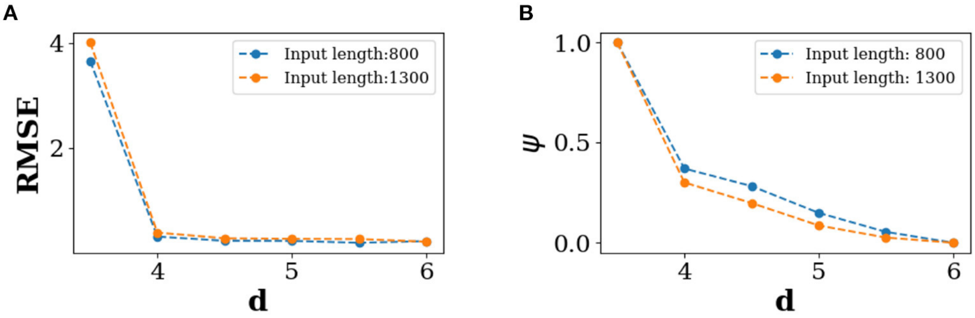


FIGURE 5
Predictability of extreme events. For 20 extreme events, **(A)** RSME against threshold d for different length of input data, **(B)** average phase against threshold d for different length of input data. Here, for both cases the average of 400 realizations are presented. Instantaneous phase $\phi_i(t)$ of *ith* signal is estimated using the Hilbert transform [67].

for higher values of d with the formation of a coherent bundle of the input signals. This indicates that there is a higher tendency of phase coherence between input signals for higher magnitude EEs which enhances the ability of the machine to predict their precursory structure.

Conclusion and discussion

We have proposed an Echo State Network based scheme for the prediction of the preceding shape of extreme events from a passive variable which shows no visible manifestation of extreme events, but connected to an active variable that has clear indications of rare and recurrent high amplitude events. Such a situation occurs in the real world where maintaining data records of subsidiary variable is easier, and may be useful for studies related to prediction of extreme events in another observable that is difficult to record. To test our scheme, we generated data using a synaptically (chemical) coupled model of two Hindmarsh-Rose (HR) neurons. Two types of variables are involved in the HR model, two fast variables (defined as active) that exhibit extreme events in their time evolution, and a slow variable (defined here as passive) having a slower time-scale and most importantly, showing no visible signs of extremes. The passive variable was considered as our input candidate for the machine for the purpose of predicting the preceding structure of extreme events in the active variable.

Our strategy was first to identify the extreme events in a long time series of an active variable with a choice of an appropriate threshold height and collect data from the passive variable that corresponds to each extreme in the active variable. We saved the data only prior to the arrival of extreme events barring all extremes, then a part of the collected dataset from the passive variable is used for testing a multi-input machine and another part of the data for testing/predicting the prior structure of the forthcoming extremes. Our results indicated that higher the magnitude of extreme events, the efficiency of the machine to predict its precursory structure is higher. Higher intensity events are defined only by increasing the threshold height. On further investigation, we found that for higher intensity extreme events the input signals collectively form a coherent pattern, which aided the machine to predict the prior structure with increased efficiency. Thus, coherence of the multi-input time signals is the key to a better prediction of the forthcoming extreme events by the machine. A possible quantitative explanation of the enhanced predictability is provided. For this purpose, a new coherence measure ψ is introduced to represent the average phase differences between the segmented time signals. It was observed that ψ decreases with increasing threshold height, therefore confirming our finding that the enhanced ability of the machine to predict higher amplitude extreme events is related to an increase in the phase coherence of the input signals.

Our machine learning scheme opens up an alternative strategy for predicting extreme events from passive variables in the real world. Furthermore, our findings maintains those reported by [37, 38] that higher the magnitude of extreme events, higher is the predictability skill. Finding suitable passive variables for real world systems is a challenge. Most of the time they typically belong to very high dimensional system and often can be a combination of multiple variables. For example, Moon and Ha [68] identified the relation between the onset of Indian summer monsoon with the soil moisture in the Iranian desert, our method could be used to predict the early warning or precursory signal to the forthcoming climate extreme if we can identify the slow variables properly.

Data availability statement

In this study, the data has been generated by numerical simulation. Further inquiries can be directed to the corresponding author/s.

Author contributions

AB, NM, AM, and CH: conceptualization. AB: data curation, formal analysis, software, and visualization. AB, AM, CH, SD, and NM: investigation. AB and NM: methodology. AB, AM, SD, and NM: resources. NM and JK: supervision. AB and SD: writing–original draft preparation. AB, AM, CH, SD, TK, JK, and NM: writing—review and editing. All authors contributed to the article and approved the submitted version.

Funding

This research has been funded by the Deutsche Forschungsgemeinschaft (DFG) within graduate research training group GRK 2043/1 Natural risk in a changing world (NatRiskChange) at the University of Potsdam. TK and AM have been supported by the National Science Centre, Poland, OPUS Program Project No. 2018/29/B/ST8/00457. CH was supported by the INSPIRE-Faculty Grant (code: IFA17-PH193).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The reviewer DG declared a shared affiliation with the author(s) CH to the handling editor at the time of review.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fams.2022.955044/full#supplementary-material

References

1. Seneviratne S, Nicholls N, Easterling D, Goodess C, Kanae S, Kossin J, et al. *Changes in Climate Extremes and Their Impacts on the Natural Physical Environment*. (2012).

2. McPhillips LE, Chang H, Chester MV, Depietri Y, Friedman E, Grimm NB, et al. Defining extreme events: a cross-disciplinary review. *Earths Future.* (2018) 6:441–55. doi: 10.1002/2017EF000686

3. Broska LH, Poganietz WR, Vögele S. Extreme events defined–A conceptual discussion applying a complex systems approach. *Futures.* (2020) 115:102490. doi: 10.1016/j.futures.2019.102490

4. Bunde A, Eichner JF, Havlin S, Kantelhardt JW. The effect of long-term correlations on the return periods of rare events. *Phys A Stat Mech Appl.* (2003) 330:1–7. doi: 10.1016/j.physa.2003.08.004

5. Jentsch V, Kantz H, Albeverio S. In: Albeverio S, Jentsch V, Kantz H, editors. *Extreme Events: Magic, Mysteries, and Challenges*. Berlin; Heidelberg: Springer (2006).

6. Dysthe K, Krogstad HE, Müller P. Oceanic rogue waves. *Annu Rev Fluid Mech.* (2008) 40:287–310. doi: 10.1146/annurev.fluid.40.111406.102203

7. Altmann EG, Hallerberg S, Kantz H. Reactions to extreme events: moving threshold model. *Phys A Stat Mech Appl.* (2006) 364:435–44. doi: 10.1016/j.physa.2005.08.074

8. Kharif C, Pelinovsky E, Slunyaev A. *Rogue Waves in the Ocean*. Springer Science & Business Media (2008).

9. Krause SM, Börries S, Bornholdt S. Econophysics of adaptive power markets: when a market does not dampen fluctuations but amplifies them. *Phys Rev E.* (2015) 92:012815. doi: 10.1103/PhysRevE.92.012815

10. Marwan N, Kurths J. Complex network based techniques to identify extreme events and (sudden) transitions in spatio-temporal systems. *Chaos Interdiscipl J Nonlinear Sci.* (2015) 25:097609. doi: 10.1063/1.4916924

11. Ray A, Rakshit S, Basak GK, Dana SK, Ghosh D. Understanding the origin of extreme events in El Niño southern oscillation. *Phys Rev E.* (2020) 101:062210. doi: 10.1103/PhysRevE.101.062210

12. Rundle JB, Klein W. *Reduction and Predictability of Natural Disasters*. Routledge (2018).

13. Sornette D. Predictability of catastrophic events: material rupture, earthquakes, turbulence, financial crashes, and human birth. *Proc Natl Acad Sci USA.* (2002) 99(Suppl. 1):2522–9. doi: 10.1073/pnas.022581999

14. Chowdhury SN, Ray A, Dana SK, Ghosh D. Extreme events in dynamical systems and random walkers: a review. *Phys Rep.* (2022) 966:1–52. doi: 10.1016/j.physrep.2022.04.001

15. Mishra A, Leo Kingston S, Hens C, Kapitaniak T, Feudel U, Dana SK. Routes to extreme events in dynamical systems: dynamical and statistical characteristics. *Chaos Interdiscipl J Nonlinear Sci.* (2020) 30:063114. doi: 10.1063/1.5144143

16. Farazmand M, Sapsis TP. Extreme events: mechanisms and prediction. *Appl Mech Rev.* (2019) 71:050801. doi: 10.1115/1.4042065

17. Chowdhury SN, Majhi S, Ozer M, Ghosh D, Perc M. Synchronization to extreme events in moving agents. *N J Phys.* (2019) 21:073048. doi: 10.1088/1367-2630/ab2a1f

18. Chowdhury SN, Majhi S, Ghosh D. Distance dependent competitive interactions in a frustrated network of mobile agents. *IEEE Trans Netw Sci Eng.* (2020) 7:3159–70. doi: 10.1109/TNSE.2020.3017495

19. Nag Chowdhury S, Kundu S, Duh M, Perc M, Ghosh D. Cooperation on interdependent networks by means of migration and stochastic imitation. *Entropy.* (2020) 22:485. doi: 10.3390/e22040485

20. Fan J, Meng J, Ludescher J, Chen X, Ashkenazy Y, Kurths J, et al. Statistical physics approaches to the complex Earth system. *Phys Rep.* (2021) 896:1–84. doi: 10.1016/j.physrep.2020.09.005

21. Boers N, Bookhagen B, Marwan N, Kurths J, Marengo J. Complex networks identify spatial patterns of extreme rainfall events of the South American Monsoon System. *Geophys Res Lett.* (2013) 40:4386–92. doi: 10.1002/grl.50681

22. Stolbova V, Martin P, Bookhagen B, Marwan N, Kurths J. Topology and seasonal evolution of the network of extreme precipitation over the Indian subcontinent and Sri Lanka. *Nonlinear Process Geophys.* (2014) 21:901–17. doi: 10.5194/npg-21-901-2014

23. Mondal S, Mishra AK. Complex networks reveal heatwave patterns and propagations over the USA. *Geophys Res Lett.* (2021) 48:e2020GL090411. doi: 10.1029/2020GL090411

24. Agarwal A, Guntu RK, Banerjee A, Gadhawe MA, Marwan N. A complex network approach to study the extreme precipitation patterns in a river basin. *Chaos Interdiscipl J Nonlinear Sci.* (2022) 32:013113. doi: 10.1063/5.0072520

25. Ghil M, Yiou P, Hallegatte S, Malamud B, Naveau P, Soloviev A, et al. Extreme events: dynamics, statistics and prediction. *Nonlinear Process Geophys.* (2011) 18:295–350. doi: 10.5194/npg-18-295-2011

26. Coles S. *An Introduction to Statistical Modeling of Extreme Values.* London: Springer-Verlag (2001).

27. Banerjee A, Goswami B, Hirata Y, Eroglu D, Merz B, Kurths J, et al. Recurrence analysis of extreme event-like data. *Nonlinear Process Geophys.* (2021) 28:213–29. doi: 10.5194/npg-28-213-2021

28. Karnatak R, Ansmann G, Feudel U, Lehnertz K. Route to extreme events in excitable systems. *Phys Rev E.* (2014) 90:022917. doi: 10.1103/PhysRevE.90.022917

29. Ray A, Mishra A, Ghosh D, Kapitaniak T, Dana SK, Hens C. Extreme events in a network of heterogeneous Josephson junctions. *Phys Rev E.* (2020) 101:032209. doi: 10.1103/PhysRevE.101.032209

30. Ansmann G, Karnatak R, Lehnertz K, Feudel U. Extreme events in excitable systems and mechanisms of their generation. *Phys Rev E.* (2013) 88:052911. doi: 10.1103/PhysRevE.88.052911

31. Amil P, Soriano MC, Masoller C. Machine learning algorithms for predicting the amplitude of chaotic laser pulses. *Chaos Interdiscipl J Nonlinear Sci.* (2019) 29:113111. doi: 10.1063/1.5120755

32. Qi D, Majda AJ. Using machine learning to predict extreme events in complex systems. *Proc Natl Acad Sci USA.* (2020) 117:52–9. doi: 10.1073/pnas.1917285117

33. Lellep M, Prexl J, Linkmann M, Eckhardt B. Using machine learning to predict extreme events in the Hénon map. *Chaos Interdiscipl J Nonlinear Sci.* (2020) 30:013113. doi: 10.1063/1.5121844

34. Pyragas V, Pyragas K. Using reservoir computer to predict and prevent extreme events. *Phys Lett A.* (2020) 384:126591. doi: 10.1016/j.physleta.2020.126591

35. Chowdhury SN, Ray A, Mishra A, Ghosh D. Extreme events in globally coupled chaotic maps. *J Phys Complexity.* (2021) 2:035021. doi: 10.1088/2632-072X/ac221f

36. Ray A, Chakraborty T, Ghosh D. Optimized ensemble deep learning framework for scalable forecasting of dynamics containing extreme events. *Chaos Interdiscipl J Nonlinear Sci.* (2021) 31:111105. doi: 10.1063/5.0074213

37. Hallerberg S, Kantz H. How does the quality of a prediction depend on the magnitude of the events under study? *Nonlinear Process Geophys.* (2008) 15:321–31. doi: 10.5194/npg-15-321-2008

38. Hallerberg S, Kantz H. Influence of the event magnitude on the predictability of an extreme event. *Phys Rev E.* (2008) 77:011108. doi: 10.1103/PhysRevE.77.011108

39. Lukoševičius M. In: Montavon G, Orr GB, Müller KR, editors. *A Practical Guide to Applying Echo State Networks.* Berlin; Heidelberg: Springer (2012).

40. Jaeger H, Haas H. Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication. *Science.* (2004) 304:78–80. doi: 10.1126/science.1091277

41. Pathak J, Hunt B, Girvan M, Lu Z, Ott E. Model-free prediction of large spatiotemporally chaotic systems from data: a reservoir computing approach. *Phys Rev Lett.* (2018) 120:024102. doi: 10.1103/PhysRevLett.120.024102

42. Zimmermann RS, Parlitz U. Observing spatio-temporal dynamics of excitable media using reservoir computing. *Chaos Interdiscipl J Nonlinear Sci.* (2018) 28:043118. doi: 10.1063/1.5022276

43. Pathak J, Lu Z, Hunt BR, Girvan M, Ott E. Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data. *Chaos Interdiscipl J Nonlinear Sci.* (2017) 27:121102. doi: 10.1063/1.5010300

44. Lin X, Yang Z, Song Y. Short-term stock price prediction based on echo state networks. *Expert Syst Appl.* (2009) 36(3 Pt 2):7313–7. doi: 10.1016/j.eswa.2008.09.049

45. Hinaut X, Dominey PF. Real-time parallel processing of grammatical structure in the fronto-striatal system: a recurrent network simulation study using reservoir computing. *PLoS ONE.* (2013) 8: e52946. doi: 10.1371/journal.pone.0052946

46. Verstraeten D, Schrauwen B, Stroobandt D, Van Campenhout J. Isolated word recognition with the Liquid State Machine: a case study. *Inform Process Lett.* (2005) 95:521–8. doi: 10.1016/j.ipl.2005.05.019

47. Lu Z, Hunt BR, Ott E. Attractor reconstruction by machine learning. *Chaos Interdiscipl J Nonlinear Sci.* (2018) 28:061104. doi: 10.1063/1.5039508

48. Mandal S, Sinha S, Shrimali MD. Machine-learning potential of a single pendulum. *Phys Rev E.* (2022) 105:054203. doi: 10.1103/PhysRevE.105.054203

49. Lu Z, Pathak J, Hunt B, Girvan M, Brockett R, Ott E. Reservoir observers: model-free inference of unmeasured variables in chaotic systems. *Chaos Interdiscipl J Nonlinear Sci.* (2017) 27:041102. doi: 10.1063/1.4979665

50. Thiede LA, Parlitz U. Gradient based hyperparameter optimization in echo state networks. *Neural Netw.* (2019) 115:23–9. doi: 10.1016/j.neunet.2019.02.001

51. Weng T, Yang H, Gu C, Zhang J, Small M. Synchronization of chaotic systems and their machine-learning models. *Phys Rev E.* (2019) 99:042203. doi: 10.1103/PhysRevE.99.042203

52. Lymburn T, Walker DM, Small M, Jungling T. The reservoir's perspective on generalized synchronization. *Chaos Interdiscipl J Nonlinear Sci.* (2019) 29:093133. doi: 10.1063/1.5120733

53. Chen X, Weng T, Yang H, Gu C, Zhang J, Small M. Mapping topological characteristics of dynamical systems into neural networks: a reservoir computing approach. *Phys Rev E.* (2020) 102:033314. doi: 10.1103/PhysRevE.102.033314

54. Panday A, Lee WS, Dutta S, Jalan S. Machine learning assisted network classification from symbolic time-series. *Chaos Interdiscipl J Nonlinear Sci.* (2021) 31:031106. doi: 10.1063/5.0046406

55. Fan H, Kong LW, Lai YC, Wang X. Anticipating synchronization with machine learning. *Phys Rev Res.* (2021) 3:023237. doi: 10.1103/PhysRevResearch.3.023237

56. Xiao R, Kong LW, Sun ZK, Lai YC. Predicting amplitude death with machine learning. *Phys Rev E.* (2021) 104:014205. doi: 10.1103/PhysRevE.104.014205

57. Mandal S, Shrimali MD. Achieving criticality for reservoir computing using environment-induced explosive death. *Chaos.* (2021) 31:031101. doi: 10.1063/5.0038881

58. Saha S, Mishra A, Ghosh S, Dana SK, Hens C. Predicting bursting in a complete graph of mixed population through reservoir computing. *Phys Rev Res.* (2020) 2:033338. doi: 10.1103/PhysRevResearch.2.033338

59. Ghosh S, Senapati A, Mishra A, Chattopadhyay J, Dana SK, Hens C, et al. Reservoir computing on epidemic spreading: a case study on COVID-19 cases. *Phys Rev E.* (2021) 104:014308. doi: 10.1103/PhysRevE.104.014308

60. Roy M, Senapati A, Poria S, Mishra A, Hens C. Role of assortativity in predicting burst synchronization using echo state network. *Phys Rev E.* (2022) 105:064205. doi: 10.1103/PhysRevE.105.064205

61. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature.* (2015) 521:436–444. doi: 10.1038/nature14539

62. Zamora-Munt J, Garbin B, Barland S, Giudici M, Leite JRR, Masoller C, et al. Rogue waves in optically injected lasers: origin, predictability, and suppression. *Phys Rev A.* (2013) 87:035802. doi: 10.1103/PhysRevA.87.035802

63. Bonatto C, Endler A. Extreme and superextreme events in a loss-modulated CO 2 laser: nonlinear resonance route and precursors. *Phys Rev E.* (2017) 96:012216. doi: 10.1103/PhysRevE.96.012216

64. Hindmarsh J, Rose R. A model of the nerve impulse using two first-order differential equations. *Nature.* (1982) 296:162–4. doi: 10.1038/296162a0

65. Mishra A, Saha S, Vigneshwaran M, Pal P, Kapitaniak T, Dana SK. Dragon-king-like extreme events in coupled bursting neurons. *Phys Rev E.* (2018) 97:062311. doi: 10.1103/PhysRevE.97.062311

66. Bonatto C, Feyereisen M, Barland S, Giudici M, Masoller C, Leite JRR, et al. Deterministic optical rogue waves. *Phys Rev Lett.* (2011) 107:053901. doi: 10.1103/PhysRevLett.107.053901

67. Rosenblum MG, Pikovsky AS, Kurths J. Phase synchronization of chaotic oscillators. *Phys Rev Lett.* (1996) 76:1804. doi: 10.1103/PhysRevLett.76.1804

68. Moon S, Ha KJ. Early Indian summer monsoon onset driven by low soil moisture in the Iranian desert. *Geophys Res Lett.* (2019) 46:10568–77. doi: 10.1029/2019GL084520

frontiers | Frontiers in Applied Mathematics and Statistics

TYPE Original Research
PUBLISHED 25 October 2022
DOI 10.3389/fams.2022.1005330

Check for updates

OPEN ACCESS

EDITED BY
Lijian Jiang,
Tongji University, China

REVIEWED BY
Majid Darehmiraki,
Behbahan Khatam Alanbia University of Technology, Iran
Khalid Hattaf,
Centre Régional des Métiers de l'Education et de la Formation (CRMEF), Morocco

*CORRESPONDENCE
Mesfin Mekuria Woldaregay
msfnmkr02@gmail.com

SPECIALTY SECTION
This article was submitted to Numerical Analysis and Scientific Computation,
a section of the journal Frontiers in Applied Mathematics and Statistics

RECEIVED 28 July 2022
ACCEPTED 03 October 2022
PUBLISHED 25 October 2022

CITATION
Wondimu GM, Woldaregay MM, Dinka TG and Duressa GF (2022) Numerical treatment of singularly perturbed parabolic partial differential equations with nonlocal boundary condition.
Front. Appl. Math. Stat. 8:1005330.
doi: 10.3389/fams.2022.1005330

Numerical treatment of singularly perturbed parabolic partial differential equations with nonlocal boundary condition

Getu Mekonnen Wondimu[1], Mesfin Mekuria Woldaregay[1]*, Tekle Gemechu Dinka[1] and Gemechis File Duressa[2]

[1]Department of Applied Mathematics, Adama Science and Technology University, Adama, Ethiopia,
[2]Department of Mathematics, Jimma University, Jimma, Ethiopia

This paper presents numerical treatments for a class of singularly perturbed parabolic partial differential equations with nonlocal boundary conditions. The problem has strong boundary layers at $x = 0$ and $x = 1$. The nonstandard finite difference method was developed to solve the considered problem in the spatial direction, and the implicit Euler method was proposed to solve the resulting system of IVPs in the temporal direction. The nonlocal boundary condition is approximated by Simpsons $\frac{1}{3}$ rule. The stability and uniform convergence analysis of the scheme are studied. The developed scheme is second-order uniformly convergent in the spatial direction and first-order in the temporal direction. Two test examples are carried out to validate the applicability of the developed numerical scheme. The obtained numerical results reflect the theoretical estimate.

KEYWORDS

singularly perturbed problems, partial differential equations, reaction-diffusion, method of lines, uniform convergence, nonlocal boundary condition

1. Introduction

Differential equations that involve a small parameter in their higher order derivative term are said to be singularly perturbed problems (SPPs) or singularly perturbed differential equations (SPDEs). Many mathematical models, starting from fluid dynamics to mathematical biology, are modeled using (SPPs). For example, high Reynold's number flow in fluid dynamics, heat transport problems with large Péclet numbers, elastic vibration, etc. [1] and the references therein. Such mathematical problems are extremely difficult to solve exactly. Thus, for treating such problems numerical methods are preferable. Various scientific and engineering processes can be modeled as integral terms over the spatial domain that appear inside or outside of the boundary conditions [2, 3]. Such problems are said to be nonlocal problems. Many physical phenomena are formulated as nonlocal mathematical models. For instance, problems in thermodynamics [4], plasma physics [5], heat conduction [6], underground water flow, and populace dynamics [7] can be reduced to nonlocal problems with integral conditions. SPPs having nonlocal boundary conditions in which the highest order derivative term is multiplied by way of a small parameter are referred to as SPPs with integral boundary conditions. Such problems exhibit boundary layer phenomena wherein the solution changes. However, the numerical treatments of SPPs attract the attention of researchers due to the boundary layer behavior of the solution. Since the small parameter multiplies the highest derivative, the small regions adjoin the domain of interest's boundaries or any interior stage at which the variable quantity undergoes a very unexpected change. As a result, these problems have strong boundary layers, which ensures that there are small areas where the solution rapidly changes within very small layers near the boundary or within the problem domain [8]. Numerically treating such SPPs with nonlocal boundary conditions is a challenging task due to a very small perturbation parameter (ε).

In recent times, a class of SPPs involving nonlocal boundary conditions have been obtained great attention from scholars. To mention few of them, the authors in Bahuguna and Dabas [9], Feng et al. [10], and Li and Sun [11] studied the existence and uniqueness of a class of SPPs with nonlocal boundary conditions. The authors in Raja and Tamilselvan [12] developed a finite difference scheme for solving a class of a system of singularly perturbed reaction diffusion equations with integral boundary conditions. Debala and Duressa [13] built a uniformly convergent numerical scheme for solving SPPs with nonlocal boundary conditions. Numerical methods for solving singularly perturbed delay differential equations (SPDDEs) are considered in Sekar and Tamilselvan [14–17]. The authors developed finite difference schemes with suitable piecewise uniform Shiskin meshes. The authors in Debela and Duressa [18] used an exponentially fitted numerical scheme to solve SPDDEs of the convection-diffusion kind with nonlocal boundary conditions. Debela and Duressa [19] improved the order of accuracy for the method proposed in Debela and Duressa [18]. Kumar and Kumari [20] developed the method based on the idea of B-spline functions and an efficient numerical method on a piecewise-uniform mesh was recommended to approximate the solutions of SPPs having a delay of unit magnitude with an integral boundary condition. In the literature, only few authors considered a class of singularly perturbed parabolic partial differential equations (SPPPDEs) with integral boundary conditions. Sekar and Tamislevan [21] investigate a numerical solution for singularly perturbed delay partial differential equations (SPDPDEs) of the reaction-diffusion type with integral boundary conditions. They developed the standard finite difference on a rectangular piecewise uniform mesh for spatial discretization and a backward difference scheme in time derivative. Gobena and Duressa [22] constructed and analyzed an accurate numerical method for solving SPDPDEs with integral boundary conditions.

In general, the classical numerical methods used for solving SPDEs are not well-posed and fail to provide an exact solution when a perturbation parameter (ε) goes to zero. Therefore, it is essential to develop a numerical method that offers suitable results for small values of the perturbation parameter. As far as the researchers' knowledge, singularly perturbed parabolic partial differential equations with nonlocal boundary conditions are first being considered and have not yet been treated numerically. In this study, we investigate a uniformly convergent numerical method for solving the problem under consideration. We used the nonstandard finite difference method for space direction and the implicit Euler method for time direction.

The contents of the paper are arranged in the following manner: A brief introduction of the given problem is discussed in Section 1. In Section 2, the properties of continuous solutions are given. In Section 3, a numerical method is formulated by using the method of lines for the given problem. Stability and convergence analysis for developed numerical methods are also studied. Numerical results and discussions are given in Section 4. In Section 5, the conclusion of the paper is given.

Notation: In this paper, N and M denote the number of mesh intervals in spatial and temporal discretization, respectively. C is a generic positive constant independent of ε, N, and M. The norm used for studying the convergence of numerical solutions is the maximum norm defined as $\left|\left|z(s,t)\right|\right| := \sup\left|z(s,t)\right|, (s,t) \in D$.

2. Properties of continuous problem

In this paper, we consider a class of singularly perturbed 1D parabolic partial differential equations of the reaction-diffusion type with non-local boundary conditions,

$$\begin{cases} \mathcal{L}z(s,t) = \left(-\varepsilon\frac{\partial^2}{\partial s^2} + \frac{\partial}{\partial t} + a(s,t)\right)z(s,t) = f(s,t) \;\; (s,t) \in D, \\ z(s,0) = \phi_b(s), \;\; \phi_b(s,t) \in \Gamma_b = \left\{(s,0)\right\}, \\ z(0,t) = \phi_l(t), \;\; \phi_l(s,t) \in \Gamma_l = \left\{(0,t); 0 \le t \le T\right\}, \\ \mathcal{K}z(s,t) = z(1,t) - \varepsilon\int_0^1 g(s)z(s,t)ds = \phi_r(s,t), \;\; \phi_r(s,t) \in \\ \qquad \Gamma_r = \left\{(1,t); 0 \le t \le T\right\}. \end{cases} \tag{1}$$

where $(s,t) \in D = \Omega_x \times \Omega_t = (0,1) \times (0,T]$, $\bar{D} = [0,1] \times [0,T]$, and ε is a small parameter ($0 < \varepsilon \ll 1$). Suppose that $a(s,t) \ge \alpha > 0$, $f(s,t)$, ϕ_l, ϕ_r, ϕ_b are sufficiently smooth functions and $g(s)$ is nonnegative monotone function and satisfies $\int_0^1 g(s)ds < 1$. The existence and uniqueness of the problem (1) can be established under the assumption that the data are Hölder continuous and imposing proper compatibility conditions at the corners [23]. Note that ϕ_l and ϕ_r are only functions of t, while ϕ_b is a function of x only. The problems necessarily satisfies the following sufficient compatibility conditions $\phi_b(0,0) = \phi_l(0,0)$, $\phi_b(1,0) =$

$\phi_r(1,0)$, and

$$-\varepsilon\frac{\partial^2\phi_b(0,0)}{\partial s^2}+a(0,0)\phi_b(0,0)+\frac{\partial\phi_l(0,0)}{\partial t}=f(0,0),$$
$$-\varepsilon\frac{\partial^2\phi_b(1,0)}{\partial s^2}+a(1,0)\phi_b(1,0)+\frac{\partial\phi_r(1,0)}{\partial t}=f(1,0).$$

Note that ϕ_l,ϕ_r, and ϕ_b are assumed to be sufficiently smooth for Equation (1) to make sense, namely $\phi_l,\phi_r\in C^1([0,T])$, and $\phi_b\in C^{(2,1)}(\Gamma_b)$.

Next, we analyze some properties of the continuous solution (Equation 1) which guarantee the existence and uniqueness of the analytical solution. A replication of this belonging in semi-discrete form can be used to present the approximate solution, which we provide in the following section.

Lemma 1. (Continuous Maximum Principle) Let $\Psi(s,t)\in C^{(0,0)}(\bar{D})\cap C^{(1,0)}(D)\cap C^{(2,1)}(D)$ be a sufficiently smooth function such that $\Psi(0,t)\geq 0, \Psi(s,0)\geq 0, \mathcal{K}\Psi(1,t)\geq 0, \mathcal{L}\Psi(s,t)\geq 0, \forall(s,t)\in D$. Then $\Psi(s,t)\geq 0$, $\forall(s,t)\in\bar{D}$, where $\mathcal{L}\Psi(s,t)=\Psi_t(s,t)-\varepsilon\Psi_{ss}(s,t)+a\Psi(s,t)$.

Proof. Assume (s^*,t^*) be defined as $\Psi(s^*,t^*)=\min_{(s,t)\in\bar{D}}\Psi(s,t)$ and suppose that $\Psi(s^*,t^*)\leq 0$. It is known that $(s^*,t^*)\notin\partial D$. Thus, $\mathcal{L}\Psi(s^*,t^*)=\Psi_t(s^*,t^*)-\varepsilon\Psi_{ss}(s^*,t^*)+a(s,t)\Psi(s^*,t^*)$. Since $\Psi(s^*,t^*)=\min_{(s,t)\in\bar{D}}\Psi(s,t)$, which indicates that $\Psi(s^*,t^*)=0$, $\Psi_t(s^*,t^*)=0$, $\Psi_{ss}(s^*,t^*)\geq 0$ and implies that $\mathcal{L}\Psi(s^*,t^*)<0$, which is contradicts with the above assumption. $\mathcal{L}\Psi(s^*,t^*)>0, \forall s\in D$. So that, $\Psi(s,t)\geq 0$, $\forall(s,t)\in D$. □

Lemma 2. (Stability Result) Assume $z(s,t)$ is the solution to the continuous problem in Equation (1). Then we have the bound

$$z(s,t)\leq\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s),\max\left\{\phi_l(s,t),\phi_r(s,t)\right\}\right\},$$

where $\left|\left|f\right|\right|=\max\left\{f(s,t)\right\}$.

Proof. We prove this by using the maximum principle Lemma (1) and by constructing the barrier functions $\theta^{\pm}(s,t)=CM\pm z(s,t),(s,t)\in\bar{D}$, where $M=\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s),\max\left\{\phi_l(s,t),\phi_r(s,t)\right\}\right\}$. At initial, we have

$$\begin{aligned}\theta^{\pm}(s,0)&=\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s),\max\left\{\phi_l(s,0),\phi_r(s,0)\right\}\right\}\\&\quad\pm z(s,0)\\&=\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s)\right\}\pm\phi_b(s)\geq 0.\end{aligned}$$

For $x=0$, we have

$$\begin{aligned}\theta^{\pm}(0,t)&=\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(0),\max\left\{\phi_l(0,t),\phi_r(0,t)\right\}\right\}\\&\quad\pm z(0,t)\\&=\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_l(t)\right\}\pm\phi_l(t)\geq 0.\end{aligned}$$

For $x=1$, we have

$$\begin{aligned}\mathcal{K}\theta^{\pm}(1,t)&=\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(1),\max\left\{\phi_l(1,t),\mathcal{K}\phi_r(1,t)\right\}\right\}\\&\quad\pm\mathcal{K}z(1,t)\\&=\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_r(1,t)\right\}\pm\phi_r(1,t)\geq 0.\end{aligned}$$

For $0<s<1$, we have

$$\begin{aligned}&\mathcal{L}\theta^{\pm}(s,t)\\&=\theta^{\pm}_t(s,t)-\varepsilon\theta^{\pm}_{ss}(s,t)+a(s,t)\theta^{\pm}(s,t),\\&=\left[\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s),\max\left\{\phi_l(s,t),\phi_r(s,t)\right\}\right\}\pm z(s,t)\right]_t\\&-\varepsilon\left[\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s),\max\left\{\phi_l(s,t),\phi_r(s,t)\right\}\right\}\pm z(s,t)\right]_{ss}\\&+a(s,t)\left(\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s),\max\left\{\phi_l(s,t),\phi_r(s,t)\right\}\right\}\pm z(s,t)\right)\\&=\max\left\{\phi_{lt}(s,t),\phi_{rt}(s,t)\right\}\pm z_t(s,t)-\varepsilon\max\left\{\phi_{bss}(s),\phi_{lss}(s,t),\phi_{rss}(s,t)\right\}\\&\pm-\varepsilon u_{ss}(s,t)+\alpha\alpha^{-1}\left|\left|f\right|\right|+\alpha\max\left\{\phi_b(s),\max\left\{\phi_l(s,t),\phi_r(s,t)\right\}\right\}\\&\pm\alpha z(s,t)\\\geq&\ 0,\end{aligned}$$

where $\varepsilon>0, a(s,t)\geq\alpha>0$. This implies that $\mathcal{L}\theta^{\pm}(s,t)\geq 0$. Hence, by Lemma 1, we have, $\theta^{\pm}(s,t)\geq 0, \forall(s,t)\in\bar{D}$, which indicates

$$z(s,t)\leq\alpha^{-1}\left|\left|f\right|\right|+\max\left\{\phi_b(s),\max\left\{\phi_l(s,t),\phi_r(s,t)\right\}\right\}.$$ □

The sufficient conditions for the existence of a unique solution is given in Lemma 3 and Theorem 1.

Lemma 3. If the coefficient satisfies $a(s,t),f(s,t)\in C^0(\bar{D})$ and boundary conditions satisfies $\phi_l\in C^1([0,T]),\phi_b\in C^{(2,1)}(\Gamma_b),\phi_r\in C^1([0,T])$ and suppose that the compatibility conditions are satisfied. Then, the problem (Equation 1) has a unique solution $z(s,t)$ which is satisfy $z(s,t)\in C^{(2,1)}(\bar{D})$.

Proof. Refer to Ladyženskaja et al. [23] □

To estimate the error for the fitted numerical technique below, the idea that the solution of Equation (1) is more regular than the one guaranteed by using the result in Theorem 1. To attain this greater regularity, stronger compatibility conditions are imposed at the corners.

Theorem 1. If the coefficient satisfies $a(s,t),f(s,t)\in C^{(2,1)}(\bar{D})$ and boundary conditions satisfies $\phi_l\in C^2([0,T]),\phi_b\in C^{(4,2)}(\Gamma_b),\phi_r\in C^2([0,T])$, Then the problem (Equation 1) having a unique solution z which satisfies $z\in C^{(4,2)}(\bar{D})$. And also the derivatives of solution z are bounded, $\forall i,j\in\mathbf{Z}\geq 0$ such that $0\leq i+2j\leq 4$,

$$\left|\left|\frac{\partial^{i+j}z}{\partial s^i\partial t^j}\right|\right|\leq C\varepsilon^{\frac{-i}{2}}.$$

Proof. The boundedness of the solutions and its derivative is given as follows. Under the stretched transformation $\tilde{s}=\frac{s}{\sqrt{\varepsilon}}$ problem (Equation 1) can be rewritten as

$$\begin{cases}\mathcal{L}\tilde{z}(\tilde{s},t)=\left(-\varepsilon\frac{\partial^2}{\partial\tilde{s}^2}+\frac{\partial}{\partial t}+\tilde{a}(\tilde{s},t)\right)\tilde{z}(\tilde{s},t)=f(\tilde{s},t), & (\tilde{s},t)\in\tilde{D}_{\varepsilon}\\ \tilde{z}(\tilde{s},t)=\phi_l(\tilde{s},t), & (\tilde{s},t)\in\tilde{\Gamma}_l\\ \mathcal{K}\tilde{z}(\tilde{s},t)=\tilde{z}(1,t)-\varepsilon\int_0^1 g(s)\tilde{z}(\tilde{s},t)ds=\phi_r(\tilde{s},t), & (\tilde{s},t)\in\tilde{\Gamma}_r\\ \tilde{z}(\tilde{s},t)=\phi_b(\tilde{s},t), & (\tilde{s},t)\in\tilde{\Gamma}_b\end{cases}\qquad(2)$$

where $\tilde{D}_\varepsilon = (0, \frac{1}{\sqrt{\varepsilon}}) \times (0, T)$, and the boundary condition $\tilde{\Gamma}$ to Γ, where Equation (2) is independent of ε. Then, by taking the idea of estimation (10.6) of Ladyženskaja et al. [23] (p. 352), we will obtain

$$\left|\left|\frac{\partial^{i+j}\tilde{z}}{\partial\tilde{s}^i \partial t^j}\right|\right|_{\tilde{N}_\delta} \leq C\left(1 + ||\tilde{z}||_{\tilde{N}_{2\delta}}\right),$$

$\forall\ \tilde{N}_\delta$ in $\tilde{D}_\varepsilon$. Here, $\tilde{N}_\delta$, $\delta > 0$ is a neighborhood with diameter δ in $\tilde{D}_\varepsilon$. Returning to the original variable

$$\left|\left|\frac{\partial^{i+j}z}{\partial\tilde{s}^i \partial t^j}\right|\right|_{\bar{D}} \leq C\varepsilon^{\frac{-i}{2}}\left(1 + ||z||_{\bar{D}}\right).$$

Hence, the proof is complete by using the bound on z of Lemma 2. □

The bounds of the derivatives of the solution given in Theorem 1 were derived from classical results. They are not adequate for the proof of the ε -uniform error estimate. Stronger bounds on these derivatives are now obtained by a method originally devised in Shishkin [24]. The main idea is to decompose the solution z into smooth and singular components.

Lemma 4. If the coefficient satisfies $a(s,t), f(s,t) \in C^{(4,2)}(\bar{D})$, and the boundary conditions satisfies $\phi_l \in C^{(3)}([0,T]), \phi_b \in C^{(6,3)}(\Gamma_b), \phi_r \in C^{(3)}([0,T])$. Then we have

$$\begin{aligned}
\|\frac{\partial^{i+j}v}{\partial s^i \partial t^j}\|_{\bar{D}} &\leq C\left(1 + \varepsilon^{1-i/2}\right), \quad (s,t) \in D\\
|\frac{\partial^{i+j}w_l}{\partial s^i \partial t^j}| &\leq C\varepsilon^{\frac{-i}{2}} e^{\frac{s}{\sqrt{\varepsilon}}}, \qquad (s,t) \in D\\
|\frac{\partial^{i+j}w_r}{\partial s^i \partial t^j}| &\leq C\varepsilon^{\frac{-i}{2}} e^{\frac{-(1-s)}{\sqrt{\varepsilon}}}, \qquad (s,t) \in D
\end{aligned}$$

where C is a constant independent of parameter ε, $(s,t) \in \bar{D}$, $i,j \geq 0, 0 \leq i + 2j \leq 4$.

Proof. For proof, the interested reader can refer to Elango et al. [21]. □

3. Numerical scheme

3.1. Spatial semi-discretization

The fundamental idea of non-standard discrete modeling techniques is the development of the exact finite difference technique. Micken presented methods and rules for developing nonstandard FDMs for various types of problems [25]. To develop a discrete scheme in keeping with Mickens' guidelines, the denominator characteristic for the discrete derivatives should be described in terms of more complicated functions with larger step sizes than those used in classical methods. These complicated functions are a general property of the method that may be useful when constructing dependable methods for such problems.

To construct a genuine finite difference scheme for the problem of the form in Equation (1), we use the methods described in Woldaregay and Duressa [26]. The constant coefficient given in Equation (3) without the time variable is considered as follows.

$$-\varepsilon\frac{d^2z(s)}{ds^2} + az(s) = 0. \tag{3}$$

By solving Equation (3), we obtain two independent solutions $e^{\mu_1 s}$ and $e^{\mu_2 s}$, where

$$\mu_{1,2} = \pm\sqrt{\alpha/\varepsilon}.$$

The spatial domain $[0,1]$ is discretized on uniform mesh length $\Delta s = h$ as follows. $D^N = \left\{s_i = s_0 + ih,\ i = 1(1)N, s_0 = 0, s_N = 1, h = 1/N\right\}$, N is taken as number of mesh points in the spatial discretization. The approximate solution of $z(s_i)$ will be denoted by Z_i. Here, the main aim is to compute difference equations that have similar results with the problem (Equation 1) at the mesh point s_i which is given by $Z_i = B_1 e^{\mu_1 s_i} + B_2 e^{\mu_2 s_i}$. Applying the procedures given in Mickens [25], we get

$$\det\begin{bmatrix} Z_{i-1} & \exp\left(\mu_1 s_{i-1}\right) & \exp\left(\mu_2 s_{i-1}\right)\\ Z_i & \exp\left(\mu_1 s_i\right) & \exp\left(\mu_2 s_i\right)\\ Z_{i+1} & \exp\left(\mu_1 s_{i+1}\right) & \exp\left(\mu_2 s_{i+1}\right)\end{bmatrix} = 0. \tag{4}$$

After simplification, Equation (4) becomes

$$Z_{i-1} - 2\cosh\left(\sqrt{\frac{\alpha}{\varepsilon}}h\right)Z_i + Z_{i+1} = 0. \tag{5}$$

which is an exact difference scheme for Equation (3). By performing some arithmetic manipulation and making rearrangement on Equation (5) for the variable coefficient problem, we obtain

$$-\varepsilon\frac{Z_{i+1} - 2Z_i + Z_{i-1}}{\lambda_i^2} + a_i Z_i = 0. \tag{6}$$

The denominator function λ_i^2 becomes

$$\lambda_i^2 = \frac{4}{\beta_i^2}\sinh^2\left(\frac{\beta_i}{2}h\right), \tag{7}$$

where λ^2 is a function of ε, β_i, h, and $\beta_i = \sqrt{\frac{a_i}{\varepsilon}}$.

For more information about nonstandard finite difference methods for reaction diffusion problems, an interested reader can refer to the study of Munyakazi and Patidar [27].

By using Equation (7), and applying the nonstandard finite difference method to a semi-discrete problem, we have

$$\frac{dZ_i(t)}{dt} - \varepsilon\frac{Z_{i+1}(t) - 2Z_i(t) + Z_{i-1}(t)}{\lambda_i^2(\varepsilon, h, t)} + a_i Z_i(t) = f(s_i, t). \tag{8}$$

with boundary conditions

$$\begin{cases} Z_i = \phi_{l_i}(t), \ \ i = 0, \\ Z_i = \phi_b, \qquad\qquad i = 1(1)N-1, \\ \mathcal{K}^N Z_N = Z_N 7 - \varepsilon \sum_{i=1}^{N} \dfrac{g_{i-1}Z_{i-1}^{j+1} + 4g_iZ_i^{j+1} + g_{i+1}Z_{i+1}^{j+1}}{3} \\ \qquad h = \phi_{rN}, \ i = N. \end{cases} \tag{9}$$

Here, for $i = N$, the integral boundary condition $\int_0^1 g(s)z(s)ds$ approximated by composite Simpson's integration rule.

$$\begin{aligned} &\int_0^1 g(s)z(s)ds = \\ &\frac{h}{3}\left(g(0)z(0) + g(N)z(N) + 2\sum_{i=1}^{N-1} g(s_{2i})z(s_{2i})\right. \\ &\left. + 4\sum_{i=1}^{N} g(s_{2i-1})z(s_{2i-1})\right) \\ &= \phi_r. \end{aligned} \tag{10}$$

Substituting Equation (10) in to Equation (9), we obtain

$$\begin{aligned} z(N) - \frac{h}{3}&\left(g(0)z(0) + g(N)z(N) + 2\sum_{i=1}^{N-1} g(s_{2i})z(s_{2i})\right. \\ &\left. +4\sum_{i=1}^{N} g(s_{2i-1})z(s_{2i-1})\right) = \phi_r. \end{aligned} \tag{11}$$

Equation (11) can be rewritten as

$$\begin{aligned} &-\frac{4\varepsilon h}{3}\sum_{i=1}^{N} g(s_{2i-1})z(s_{2i-1}) - \frac{2\varepsilon h}{3}\sum_{i=1}^{N-1} g(s_{2i})z(s_{2i}) \\ &+\left(1 - \frac{\varepsilon h}{3}g(N)\right)z(N) = \phi_r + \frac{\varepsilon h}{3}g(0)z(0). \end{aligned}$$

Assume that the approximation of $z(s_i, t)$ is denoted as $Z_i(t)$, by using the non-standard finite difference approximation. At this level, Equation (1) is reduced in the form of semi-discrete as follows.

$$\begin{cases} \mathcal{L}^h Z_i(t) = \frac{dZ_i(t)}{dt} \\ -\varepsilon\dfrac{Z_{i+1}(t) - 2Z_i(t) + Z_{i-1}(t)}{\lambda_i^2(\varepsilon, h, t)} + a_i Z_i(t) = f(s_i, t), \\ Z_i(0) = \phi_b(s_i), \\ Z_0(t) = \phi_l(0, t), \\ \mathcal{K}Z_N(t) = \phi_r(N, t). \end{cases} \tag{12}$$

Equation (12) is the system of IVPs and its compact form is written as

$$\frac{dZ_i(t)}{dt} + BZ_i(t) = F_i(t), \tag{13}$$

where B is $(N-1) \times (N-1)$ tridiagonal matrix, $Z_i(t)$ and $F_i(t)$ are $(N-1)$ entries of the column vector. The entries of B and F respectively given as

$$\begin{cases} b_{i,i} = \frac{2\varepsilon}{\lambda_i^2(\varepsilon,h,t)} + a(s_i), \quad i = 1(1)N-1 \\ b_{i,i-1} = -\frac{2\varepsilon}{\lambda_i^2(\varepsilon,h,t)}, \quad i = 2(1)N-1 \\ b_{i,i+1} = -\frac{2\varepsilon}{\lambda_i^2(\varepsilon,h,t)}, \quad i = 1(1)N-1, \end{cases}$$

and

$$\begin{cases} F_1(t) = f_1(t) - \left(a(s_1) + \frac{2\varepsilon}{\lambda_1^2(\varepsilon,h,t)}\right)\phi_l(0,t), \\ F_i(t) = f_i(t), \quad i = 2(1)N-1 \\ F_{N-1}(t) = f_{N-1}(t) - \frac{2\varepsilon}{\lambda_{N-1}^2(\varepsilon,h,t)}\phi_{rN}(t) \end{cases}$$

3.2. Stability and convergence analysis

Here, we present the maximum principle and uniform stability estimate of the semi-discrete operator $\mathcal{L}^h$ and its convergence analysis.

Lemma 5. (Semi-discrete Maximum Principle): Assume that $Z_0(t) \geq 0$, $\mathcal{K}Z_N(t) \geq 0$. Then $\mathcal{L}^h Z_i(t) \geq 0 \ \forall \ i = 1(1)N-1$, implies that $Z_i(t) \geq 0 \ \forall \ i = 0(1)N$.

Proof. Assume there exists $q \in \{0, \cdots, N\}$ such that $Z_q(t) = \min_{0 \leq i \leq N} Z_i(t)$. Suppose $Z_q(t) \leq 0$, which implies $q \neq 0, N$. Also, we have $Z_{q+1} - Z_q > 0$ and $Z_q - Z_{q-1} < 0$. Here, we have

$$\mathcal{L}^h Z_q(t) = \frac{dZ_q(t)}{dt} - \varepsilon\frac{Z_{q+1}(t) - 2Z_q(t) + Z_{q-1}(t)}{\lambda_q^2} + a_q Z_q(t).$$

By using the above assumption, we get that $\mathcal{L}^h Z_i(t) < 0$, for $i = 1(1)N-1$. Thus, the assumption $Z_i(t) < 0$, $i = 0(1)N$ is not correct. Hence, $Z_i(t) \geq 0 \ \forall \ i = 0(1)N$. □

This Lemma 5 is used to obtain the bounds of the discrete solution given in Lemma 6. In general, the discrete maximum principle is widely used to show the boundedness and positivity of a discrete solution.

Lemma 6. The solution $Z_i(t)$ of the semidiscrete problem in Equations (12) or (13) satisfies the following bound.

$$\left|Z_i(t)\right| = \frac{1}{\alpha}\max_i\left|\mathcal{L}^h Z_i(t)\right| + \max_i\left\{\phi_b(s_i), \max_i\left\{\phi_l(s_i,t), \phi_r(s_i,t)\right\}\right\}.$$

Proof. Suppose $q = \frac{1}{\alpha}\max_i\left|\mathcal{L}^h Z_i(t)\right| + \max_i\left\{\phi_b(s_i), \max_i\left\{\phi_l(s_i,t), \phi_r(s_i,t)\right\}\right\}$ and define a comparison function $\theta_i^{\pm}(t)$ as

$$\theta_i^{\pm}(t) = q \pm Z_i(t).$$

For the points on the boundary, we have

$$\theta_0^{\pm}(t) = q \pm Z_0(t) = q \pm \phi_l(0,t) \geq 0,$$
$$\mathcal{K}^N\theta_N^{\pm}(t) = q \pm \mathcal{K}^N Z_N(t) = q \pm \phi_r(1,t) \geq 0.$$

For $1 \leq i \leq N-1$, we have

$$\begin{aligned}\mathcal{L}^h\theta_i^{\pm}(t) \ &= \ \frac{d(q \pm Z_i(t))}{dt}\\ &- \ \varepsilon\frac{\big(q \pm Z_{i-1}(t) - 2(q \pm Z_i(t)) + q \pm Z_{i+1}(t)\big)}{\lambda^2}\\ &+ \ a_i(q \pm Z_i(t))\\ &= \ a_iq \pm \mathcal{L}^h Z_i(t)\\ &= \ a_i\left(\alpha^{-1}\max_i\left|\mathcal{L}^h Z_i(t)\right|\right.\\ &+ \ \left.\max_i\left\{\phi_b(s_i), \max_i\left\{\phi_l(s_i,t), \phi_r(s_i,t)\right\}\right\}\right) \pm f_{i,j}\\ &\geq \ 0, \qquad \text{since } \ a_i \geq \alpha.\end{aligned}$$

From Lemma 5, we get, $\theta_i^{\pm}(t) \geq 0, \forall\ (s_i,t) \in \bar{\Omega}_x^N \times (0,T)$. □

Next, we present the convergence analysis of spatial discretization. We denoted $Z_i(t)$ as approximate solution for the spatial semidiscretization to the exact solution $z(s,t)$ at $s = s_i$, $i = 0(1)N$. Let us define the backward and forward finite differences in space as:

$$D^-z(s_i,t) = \frac{z(s_i,t) - z(s_{i-1})}{h}, \ D^+z(s_i,t) = \frac{z(s_{i+1},t) - z(s_i,t)}{h},$$

respectively, and the second order central finite difference operator as

$$\delta^2 z(s_i,t) = D^+D^-z(s_i,t) = \frac{D^+z(s_i,t) - D^-z(s_i,t)}{h}.$$

Lemma 7. Let N be a fixed mesh. Then, for $\varepsilon \to 0$, we have

$$\lim_{\varepsilon\to 0}\max_{1\leq i\leq N-1}\frac{\exp(-ps_i/\sqrt{\varepsilon})}{\varepsilon^{m/2}} = 0 \quad \text{and}$$
$$\lim_{\varepsilon\to 0}\max_{1\leq i\leq N-1}\frac{\exp(-p(1-s_i)/\sqrt{\varepsilon})}{\varepsilon^{m/2}} = 0.$$

where $m = 1, 2, 3, \cdots$.

Proof. Refer to Munyakazi and Patidar [27] □

Theorem 2. Let the coefficient function $a(s)$ and the function $f(s,t)$ in Equation (12) be sufficiently smooth and $z(s,t) \in C^4\left(\bar{D}\right)$. Then the semidiscrete solution $Z_i(t)$ of Equation (12) satisfies

$$\left|\mathcal{L}^h\left(z(s_i,t) - Z_i(t)\right)\right| \leq Ch^2.$$

Proof. The truncation error in spatial direction is considered as

$$\begin{aligned}&\mathcal{L}^h\left(z(s_i,t) - Z_i(t)\right) = \mathcal{L}^h z(s_i,t) - \mathcal{L}^h Z_i(t)\\ &= -\varepsilon\frac{d^2}{ds^2}z(s_i,t) + \frac{D_s^+D_s^+h^2}{\lambda^2}z(s_i,t)\\ &= -\varepsilon\frac{d^2}{ds^2}z(s_i,t) + \frac{\varepsilon}{\lambda^2}\left(h^2\frac{d^2}{ds^2}z(s_i,t) + \frac{h^4}{12}\frac{d^4}{ds^4}z(s_i,t)\right).\end{aligned} \tag{14}$$

Note that we have used Taylor expansions of $z_{i-1}(t)$ and $z_{i+1}(t)$. A truncated Taylor expansion of $\frac{1}{\lambda^2}$ of order five becomes

$$\frac{1}{\lambda^2} = \frac{\beta^2}{4}\left(\frac{4}{\beta^2h^2} - \frac{1}{3} + \frac{\beta^2h^2}{60}\right). \tag{15}$$

Using Equation (15) in Equation (14), we obtain

$$\begin{aligned}\mathcal{L}^h\left(z(s_i,t) - Z_i(t)\right) &= \frac{\varepsilon}{12}\left(\frac{d^4}{ds^4}z(s_i,t) - \beta^2\frac{d^2}{ds^2}z(s_i,t)\right)h^2\\ &+ \varepsilon\beta^2h^4\left(\frac{\beta^2}{240}\frac{d^2z(s_i,t)}{ds^2} - \frac{1}{144}\frac{d^4z(s_i,t)}{ds^4}\right) + h^6\frac{\varepsilon\beta^4}{2880}\frac{d^4z(s_i,t)}{ds^4}.\end{aligned} \tag{16}$$

We use Lemma (7), to obtain the boundedness of Equation (16). Using Lemma (7) and Theorem (1), we obtain

$$\left|\mathcal{L}^h\left(z(s_i,t) - Z_i(t)\right)\right| \leq CN^{-2}.$$

The truncation error at $s = s_N$, become

$$\begin{aligned}&\mathcal{K}^N\left(Z(s_i) - z(s_i)\right) = \mathcal{K}^N Z(s_N) - \mathcal{K}^N s(s_i),\\ &= \phi_r - \mathcal{K}^N Z(s_N),\\ &= \mathcal{K}z(s_i) - \mathcal{K}^N Z(s_N),\\ &= z(s_N) - \varepsilon\int_0^1 g(s)z(s)ds - \left(Z(s_N) - \varepsilon\int_{s_0}^{s_N} g(s)z(s)ds\right),\\ &= \varepsilon\int_{s_0}^{s_N} g(s)z(s)ds - \varepsilon\sum_{i=1}^{N}\frac{g_{i-1}z_{i-1} + 4g_iz_i + g_{i+1}z_{i+1}}{3}h,\\ &= \varepsilon\left[\int_{s_0}^{s_1} g(s)z(s)ds + \int_{s_1}^{s_2} g(s)z(s)ds + \cdots\right.\\ &+ \left.\int_{s_N}^{s_{N+1}} g(s)z(s)ds\right]\\ &-\varepsilon\left[\frac{g_0z_0 + 4g_1z_1 + g_2z_2}{3}h + \cdots\right.\\ &+ \left.\frac{g_{N-1}z_{N-1} + 4g_Nz_N + g_{N+1}z_{N+1}}{3}h\right],\\ &\left|\mathcal{K}^N\left(Z(s_i) - z(s_i)\right)\right|\\ &= \left|C\varepsilon\left(h^4z^{(4)}(\xi_1) + h^4z^{(4)}(\xi_2) + \cdots + h^4z^{(4)}(\xi_N)\right)\right|,\\ &\left|\mathcal{K}^N\left(Z(s_i) - z(s_i)\right)\right|\\ &\leq C\varepsilon h^4\left(z^{(4)}(\xi_1) + z^{(4)}(\xi_2) + \cdots + z^{(4)}(\xi_N)\right),\\ &\leq \ C\varepsilon h^4\left\|\left|\frac{d^4z(\xi_i)}{dx^4}\right|\right\| \leq Ch^2 = CN^{-2}.\end{aligned} \tag{17}$$

□

Theorem 3. The semidiscrete solutions satisfy the uniform error bound

$$\sup_{0<\varepsilon\ll 1}\max_{i}\left|z(s_i,t)-Z_i(t)\right|_{\bar{D}}\leq CN^{-2}. \tag{18}$$

Proof. The proof follows from Theorem (1) and Lemma (7) under the properties of boundedness of a semi-discrete solution and the required bound is satisfied.

3.3. Temporal discretization

A mesh with length $\Delta t = t_{j+1}-t_j, j = 0(1)M-1$ is constructed on the time domain $D_t = [0,T]$, where M is a positive integer. The IVPs Equation (13) are discretized using the implicit Euler method on a uniform mesh. By denoting the approximation of $z_i(t_j)$ as Z_i^j, we construct the time discretization as follows.

$$\frac{Z_i^j-Z_i^{j-1}}{\Delta t}=BZ_i^j+F_i^j \tag{19}$$

with the initial condition $Z_0(t) = \phi_l(t_j)$, and by rearranging Equation (19), we obtain

$$Z_i^j=[I+\Delta tB]^{-1}\left[\Delta tF_i^j+Z_i^{i-1}\right]. \tag{20}$$

Lemma 8. Suppose $\left|\frac{\partial^i z(s,t_j)}{\partial t^i}\right|\leq C,\ \forall(s,t)\in\bar{D},\ i=0,1,2.$ Then the local truncation error associated with the time direction satisfies $|e_j|\leq C(\Delta t)^2$.

Proof. The local truncation error is defined as

$$\begin{aligned}e_j &= z(t_j)-Z_i^j\\ &= z(t_j)-[I+\Delta tB]^{-1}\left[\Delta tF_i^j+Z_i^{j-1}\right].\end{aligned}$$

Using Taylor expansion, we obtain $z(t_{j-1})$ as

$$z(t_{j-1})=z(t_j)-\Delta tz_t(t_j)+\frac{(\Delta t)^2}{2}z_{tt}(t_j)+\frac{(\Delta t)^3}{3!}z_{ttt}(t_j)+\mathcal{O}((\Delta t)^4).$$

However, $z_t(t_j)=F(t_j)-B(t_j)z(t_j)$. Thus,

$$z(t_{j-1})=z(t_j)-\Delta t[F(t_j)-B(t_j)z(t_j)]+\frac{(\Delta t)^2}{2}z_{tt}(t_j)$$
$$+\frac{(\Delta t)^3}{3!}z_{ttt}(t_j)+\mathcal{O}((\Delta t)^4).$$

Now, the local truncation error e_j becomes

$$\begin{aligned}e_j &= z(t_j)-[I+\Delta tB]^{-1}\left[\Delta tF_i^j+Z_i^{j-1}\right]\\ &= z(t_j)-[I+\Delta tB]^{-1}\left[[I+\Delta tB(t_j)]z(t_j)+\frac{(\Delta t)^2}{2}z_{tt}(t_j)+\cdots\right]\\ &= [I+\Delta tB]^{-1}\left[\frac{(\Delta t)^2}{2}z_{tt}(t_j)-\frac{(\Delta t)^3}{3!}z_{ttt}(t_j)+\mathcal{O}((\Delta t)^4)\right].\end{aligned}$$

Since the matrix B is invertible, using the relation $(\Delta t)^2 > (\Delta t)^3$ for small Δt and $z(t_j)\leq C$, we obtain

$$\begin{aligned}||e_j|| &\leq \left|\left|[I+\Delta tB]^{-1}\right|\right|\left|\left|\frac{(\Delta t)^2}{2}z_{tt}(t_j)-\frac{(v)^3}{3!}z_{ttt}(t_j)+\mathcal{O}((\Delta t)^4)\right|\right|\\ &\leq \left|\left|[I+\Delta tB]^{-1}\right|\right|(\Delta t)^2\leq C(\Delta t)^2.\end{aligned}$$

□

Lemma 9. The global error estimate in the time direction is given by $||E_{j+1}|| \leq C\Delta t, \quad \forall j \leq T/\Delta t$, where $E_{j+1} = \max_i\left|Z_i(t_{j+1})-Z_{i,j+1}\right|_D$.

Proof. The global error estimate at $(j+1)^{th}$ time step is obtained by using the local error estimate up to j^{th} time step as follows.

$$\begin{aligned}||E_{j+1}|| &= \left|\left|\sum_{i=1} e_j\right|\right| \qquad j\leq T/\Delta t\\ &\leq ||e_1||+||e_2||+||e_3||+||e_4||+\cdots+||e_j||\\ &\leq C_1(j\Delta t)\Delta t\\ &\leq C_1T\Delta t \qquad \text{since } j\Delta t\leq T\\ &\leq C\Delta t.\end{aligned}$$

Hence,

$$\left|\left|E^{j+1}\right|\right|=\max_i\left|Z_i(t_{j+1})-Z_i^{j+1}\right|_D\leq C\Delta t. \tag{21}$$

where C is a positive constant independent of ε and Δt. By taking the supremum $\forall\ \varepsilon\in(0,1]$, we obtained

$$\sup_{0<\varepsilon\ll 1}\max_i\left|Z_i(t_{j+1})-Z_i^{j+1}\right|_D\leq C\Delta t. \tag{22}$$

□

We summarizes the results of this work by considering the error estimate obtained in Equations (18) and (22) and we conclude by the following theorem.

Theorem 4. The error estimate for the solution of the continuous and fully discrete problems is given by

$$\sup_{0<\varepsilon<<1}\max_{0\leq i\leq N}\max_{0\leq i\leq M}\left|\left|z(s,t)-Z_i^{j+1}\right|\right|\leq C\left(N^{-2}+\Delta t\right),$$

where $z(s,t)$ and Z_i^{j+1} are the solutions to problems Equations (1) and (12), respectively.

Proof. The error estimation of the fully discrete scheme is given as follows.

$$\sup_{\varepsilon}\max_{i,j}\left|z(s_i,t_j)-Z_i^j\right|=\sup_{\varepsilon}\max_{i,j}\left|z(s_i,t_j)-Z_i(t_j)+Z_i(t_j)-Z_i^j\right|$$
$$\leq\sup_{\varepsilon}\max_{i,j}\left|z(s_i,t_j)-Z_i(t_j)\right|+\sup_{\varepsilon}\max_{i,j}\left|Z_i(t_j)-Z_i^j\right|.$$

Then, by combining the bound given in Theorem 3 and Lemma 9, the theorem gets proved.

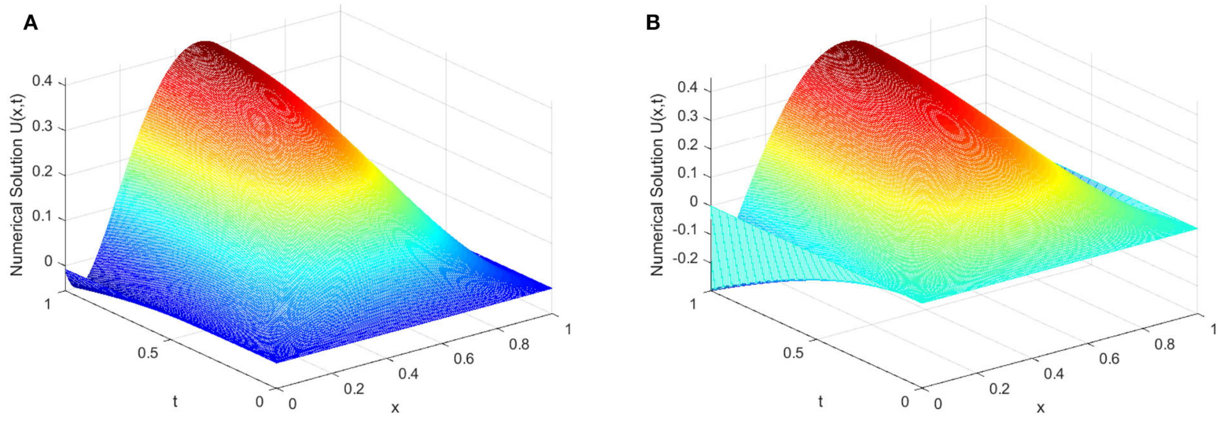


FIGURE 1
3-D graph of numerical solution for Example (1) which displays the existing layer. **(A)** $\varepsilon = 10^{-2}$. **(B)** $\varepsilon = 10^{-12}$.

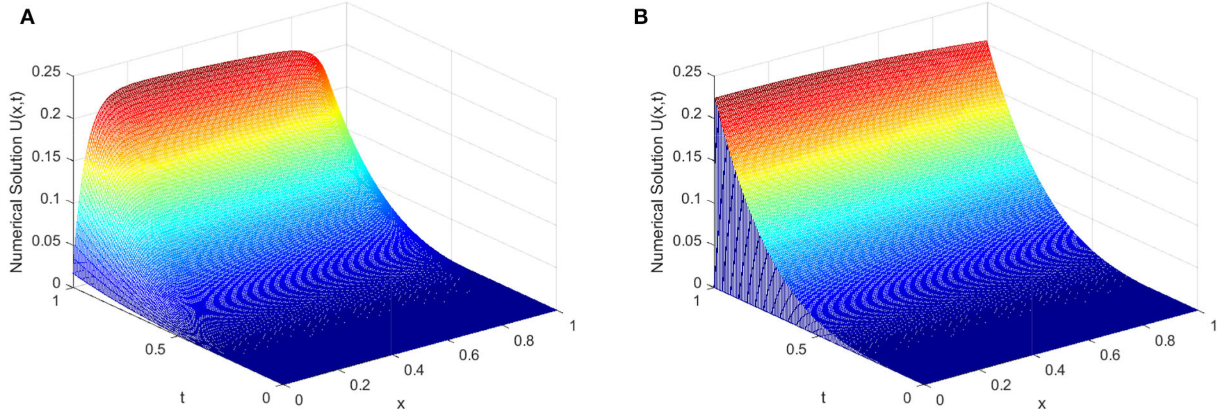


FIGURE 2
3-D graph of numerical solution for Example (2) that displays the existing layer. **(A)** $\varepsilon = 10^{-2}$. **(B)** $\varepsilon = 10^{-12}$.

4. Numerical examples, results, and discussions

Here, we developed an algorithm for the proposed method for the problem and perform experiments to validate the theoretical justifications and results. Since the exact solutions of the given examples are not known, we use double mesh techniques to obtain the maximum pointwise error of the developed scheme. Now, let $U^{N,\Delta t}$ be a conducted solution of a problem using mesh points N and time step size Δt. Again, $U_{i,j}^{2N,\Delta t/2}$ be a conducted solution on double mesh points of $2N$ and half of time step size $\Delta t/2$.

We calculate the maximum absolute error as $E_\varepsilon^{N,\Delta t} = \max_{i,j}\left|Z_{i,j}^{N,\Delta t} - Z_{i,j}^{2N,\Delta t/2}\right|$, and the parameter uniform error estimate by using $E^{N,\Delta t} = \max_\varepsilon\left(E_\varepsilon^{N,\Delta t}\right)$. We calculate the rate of convergence of the developed scheme by using $P_\varepsilon^{N,\Delta t} = \log_2(E_\varepsilon^{N,\Delta t}) - \log_2(E_\varepsilon^{2N,\Delta t/2})$. The parameter rate of convergence is calculated as $P^{N,\Delta t} = \log_2(E^{N,\Delta t}) - \log_2(E^{2N,\Delta t/2})$.

Example 1.

$$\begin{cases} \frac{\partial z(s,t)}{\partial t} - \varepsilon\frac{\partial^2 z(s,t)}{\partial s^2} + \frac{1+s^2}{2}z(s,t) = e^{-t} - 1 \\ + \sin(\pi\ s), \quad (s,t) \in (0,1) \times (0,1] \\ z(s,0) = 0, \qquad\qquad\qquad\qquad s \in (0,1), \\ z(0,t) = 0, \qquad\qquad\qquad\qquad t \in (0,1], \\ \mathcal{K}z(1,t) = z(1,t) - \varepsilon\int_0^1 \frac{s}{6}z(s,t)ds = 0, \quad t \in (0,1]. \end{cases}$$

TABLE 1 Maximum absolute error and rate of convergence of the scheme for Example (1).

ε	$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$
$\downarrow$	$\Delta t = 0.1$	$\Delta t = 0.1/4$	$\Delta t = 0.1/4^2$	$\Delta t = 0.1/4^3$	$\Delta t = 0.1/4^4$
10^{-6}	1.2294e-02	3.3054e-03	8.4694e-04	2.1381e-04	5.3681e-05
	1.8951	1.9645	1.9859	1.9938	-
10^{-8}	1.2294e-02	3.3054e-03	8.4694e-04	2.1381e-04	5.3681e-05
	1.8951	1.9645	1.9859	1.9938	-
10^{-10}	1.2294e-02	3.3054e-03	8.4694e-04	2.1381e-04	5.3681e-05
	1.8951	1.9645	1.9859	1.9938	-
10^{-12}	1.2294e-02	3.3054e-03	8.4694e-04	2.1381e-04	5.3681e-05
	1.8951	1.9645	1.9859	1.9938	-
10^{-14}	1.2294e-02	3.3054e-03	8.4694e-04	2.1381e-04	5.3681e-05
	1.8951	1.9645	1.9859	1.9938	-
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
10^{-20}	1.2294e-02	3.3054e-03	8.4694e-04	2.1381e-04	5.3681e-05
	1.8951	1.9645	1.9859	1.9938	-
$E^{N,\Delta t}$	1.2294e-02	3.3054e-03	8.4694e-04	2.1381e-04	5.3681e-05
$P^{N,\Delta t}$	1.8951	1.9645	1.9859	1.9938	-

TABLE 2 Maximum absolute error and rate of convergence of the scheme for Example (2).

ε	$N = 32$	$N = 64$	$N = 128$	$N = 256$	$N = 512$
$\downarrow$	$\Delta t = 0.1$	$\Delta t = 0.1/4$	$\Delta t = 0.1/4^2$	$\Delta t = 0.1/4^3$	$\Delta t = 0.1/4^4$
10^{-6}	1.5809e-02	5.4540e-03	1.4696e-03	3.7419e-04	9.3970e-05
	1.5354	1.8919	1.9736	1.9935	-
10^{-8}	1.5809e-02	5.4540e-03	1.4696e-03	3.7419e-04	9.3970e-05
	1.5354	1.8919	1.9736	1.9935	-
10^{-10}	1.5809e-02	5.4540e-03	1.4696e-03	3.7419e-04	9.3970e-05
	1.5354	1.8919	1.9736	1.9935	-
10^{-12}	1.5809e-02	5.4540e-03	1.4696e-03	3.7419e-04	9.3970e-05
	1.5354	1.8919	1.9736	1.9935	-
10^{-14}	1.5809e-02	5.4540e-03	1.4696e-03	3.7419e-04	9.3970e-05
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
10^{-20}	1.5809e-02	5.4540e-03	1.4696e-03	3.7419e-04	9.3970e-05
	1.5354	1.8919	1.9736	1.9935	-
$E^{N,\Delta t}$	1.5809e-02	5.4540e-03	1.4696e-03	3.7419e-04	9.3970e-05
$P^{N,\Delta t}$	1.5354	1.8919	1.9736	1.9935	-

Example 2.

$$\begin{cases} \frac{\partial z(s,t)}{\partial t} - \varepsilon \frac{\partial^2 z(s,t)}{\partial s^2} + \frac{1+s^2}{2} z(s,t) = t^3, & (s,t) \in (0,1) \times (0,1] \\ z(s,0) = 0, & s \in (0,1), \\ z(0,t) = 0, & t \in (0,1], \\ \mathcal{K}z(1,t) = z(1,t) - \varepsilon \int_0^1 \cos(s) z(s,t) ds = 0, & t \in (0,1]. \end{cases}$$

The solutions of the above two examples exhibit strong boundary layers near $x = 0$ and $x = 1$. We presented the surface plots for numerical solutions of Examples 1 and 2 in Figures 1, 2 respectively, which display the presence of boundary layers formation on the left and right side of the spatial domain for different values of ε. The maximum pointwise error and rate of convergence of the proposed schemes of Examples 1 and 2 are given in Tables 1, 2 respectively for various values of the perturbation parameter ε, mesh number N and time step size Δt. From these tables, one can observe that the developed scheme is parameter uniform convergent, which supports the theoretical results. Figure 3 indicates the Log-Log plots for the

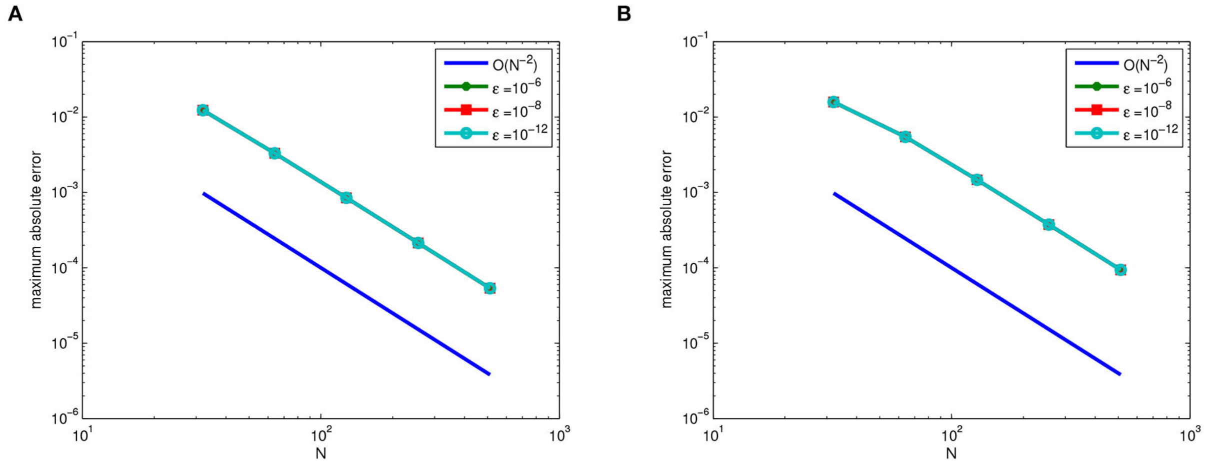


FIGURE 3
The Log-Log plot of the maximum absolute error for different values of ε for Examples 1 and 2, respectively. **(A)** Log-Log plot for Example (1). **(B)** Log-Log plot for Example (2).

maximum absolute error vs. mesh number N for the singular perturbation parameter ε. One can observe that as ε goes very small, the developed method converges uniformly independent of the perturbation parameter ε.

5. Conclusion

This paper investigates a numerical treatment for a class of singularly perturbed parabolic partial differential equations of the reaction-diffusion type with nonlocal boundary conditions. To solve the problem at hand, we employed the method of lines. A nonstandard finite difference scheme is used to semi-discretize the spatial direction, and the implicit Euler method is used to discretize the results of initial value problems. To deal with the integral boundary condition, we used a composite Simpson's $\frac{1}{3}$ rule. The stability of the evolved numerical scheme is established, and the scheme's uniform convergence is demonstrated. To validate the problem's applicability, two test examples are carried out for numerical computation for different values of the perturbation parameter ε and mesh points. The entire procedure has been demonstrated to be second-order uniformly convergent in the spatial direction and first-order in the temporal direction. The theoretical estimation is reflected in our numerical results.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Acknowledgments

Authors are grateful to their reviewers for their contributions.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Kumar M, Paul. Methods for solving singular perturbation problems arising in science and engineering. *Math Comput Model*. (2011) 54:556–75. doi: 10.1016/j.mcm.2011.02.045

2. Dehghan M, Tatari M. Use of radial basis functions for solving the second-order parabolic equation with nonlocal boundary conditions. *Math Comput Simulat*. (2008) 24:924–38. doi: 10.1002/num.20297

3. Amiraliyev G, Amiraliyeva I, Kudu M. A numerical treatment for singularly perturbed differential equations with integral boundary condition. *Appl Math Comput*. (2007) 185:574–82. doi: 10.1016/j.amc.2006.07.060

4. Day W. Extensions of a property of the heat equation to linear thermoelasticity and other theories. *Q Appl Math*. (1982) 40:319–30. doi: 10.1090/qam/678203

5. Bouziani A. Mixed problem with boundary integral conditions for a certain parabolic equation. *J Appl Math Stochastic Anal*. (1996) 9:323–30. doi: 10.1155/S1048953396000305

6. Cannon J. The solution of the heat equation subject to the specification of energy. *Q Appl Math*. (1963) 21:155–60. doi: 10.1090/qam/160437

7. Kudu M, Amiraliyev GM. Finite difference method for a singularly perturbed differential equations with integral boundary condition. *Int J Math Comput*. (2015) 26:71–9.

8. OMalley RE. Ludwig prandtls boundary layer theory. In: *Historical Developments in Singular Perturbations*. Springer (2014). p. 1–26.

9. Bahuguna D, Dabas J. Existence and uniqueness of a solution to a semilinear partial delay differential equation with an integral condition. *Nonlinear Dyn Syst Theory*. (2008) 8:7–19.

10. Feng M, Ji D, Ge W. Positive solutions for a class of boundary-value problem with integral boundary conditions in Banach spaces. *J Comput Appl Math*. (2008) 222:351–63. doi: 10.1016/j.cam.2007.11.003

11. Li H, Sun F. Existence of solutions for integral boundary value problems of second-order ordinary differential equations. *Boundary Value Problems*. (2012) 2012:1–7. doi: 10.1186/1687-2770-2012-147

12. Raja V, Tamilselvan A. Fitted finite difference method for third order singularly perturbed convection diffusion equations with integral boundary condition. *Arab J Math Sci*. (2019) 25:231–42. doi: 10.1016/j.ajmsc.2018.10.002

13. Debela, Duressa GF. Uniformly convergent numerical method for singularly perturbed convection-diffusion type problems with nonlocal boundary condition. *Int J Num Methods Fluids*. (2020) 92:1914–26. doi: 10.1002/fld.4854

14. Sekar E, Tamilselvan A. Finite difference scheme for singularly perturbed system of delay differential equations with integral boundary conditions. *J Korean Soc Ind Appl Math*. (2018) 22:201–15. doi: 10.12941/jksiam.2018.22.201

15. Sekar E, Tamilselvan A. Finite difference scheme for third order singularly perturbed delay differential equation of convection diffusion type with integral boundary condition. *J Appl Math Comput*. (2019) 61:73–86. doi: 10.1007/s12190-019-01239-0

16. Sekar E, Tamilselvan A. Singularly perturbed delay differential equations of convection-diffusion type with integral boundary condition. *J Appl Math Comput*. (2019) 59:701–22. doi: 10.1007/s12190-018-1198-4

17. Sekar E, Tamilselvan A. Third order singularly perturbed delay differential equation of reaction diffusion type with integral boundary condition. *J Appl Math Comput Mech*. (2019) 18:99–110. doi: 10.17512/jamcm.2019.2.09

18. Debela, Duressa GF. Exponentially fitted finite difference method for singularly perturbed delay differential equations with integral boundary condition. *Int J Eng Appl Sci*. (2019) 11:476–93. doi: 10.24107/ijeas.647640

19. Debela, Duressa GF. Accelerated fitted operator finite difference method for singularly perturbed delay differential equations with non-local boundary condition. *J Egypt Math Soc*. (2020) 28:1–16. doi: 10.1186/s42787-020-00076-6

20. Kumar D, Kumari P. A parameter-uniform collocation scheme for singularly perturbed delay problems with integral boundary condition. *J Appl Math Comput*. (2020) 63:813–28. doi: 10.1007/s12190-020-01340-9

21. Elango S, Tamilselvan A, Vadivel R, Gunasekaran N, Zhu H, Cao J, et al. Finite difference scheme for singularly perturbed reaction diffusion problem of partial delay differential equation with nonlocal boundary condition. *Adv Diff Equat*. (2021) 2021:1–20. doi: 10.1186/s13662-021-03296-x

22. Gobena WT, Duressa GF. Parameter-uniform numerical scheme for singularly perturbed delay parabolic reaction diffusion equations with integral boundary condition. *Int J Diff Equat*. (2021) 2021:9993644. doi: 10.1155/2021/9993644

23. Ladyženskaja OA, Solonnikov VA, Ural'ceva NN. Linear and quasi-linear equations of parabolic type. In: *American Mathematical Society, Vol. 23*. Petersburg, VA (1988).

24. Shishkin GI. Approximation of the solutions of singularly perturbed boundary-value problems with a parabolic boundary layer. *USSR Comput Math Math Phys*. (1989) 29:1–10. doi: 10.1016/0041-5553(89)90109-2

25. Mickens RE. Advances in the applications of nonstandard finite difference schemes. In: *World Scientific*. (2005).

26. Woldaregay MM, Duressa GF. Uniformly convergent numerical method for singularly perturbed delay parabolic differential equations arising in computational neuroscience. *Kragujevac J Math*. (2022) 46:65–84. doi: 10.46793/KgJMat2201.065W

27. Munyakazi JB, Patidar KC. A fitted numerical method for singularly perturbed parabolic reaction-diffusion problems. *Comput Appl Math*. (2013) 32:509–19. doi: 10.1007/s40314-013-0033-7

TYPE Review
PUBLISHED 28 October 2022
DOI 10.3389/fams.2022.1039758

Check for updates

OPEN ACCESS

EDITED BY
Jesus Manuel Munoz-Pacheco,
Benemérita Universidad Autónoma de Puebla, Mexico

REVIEWED BY
Musheer Ahmad,
Jamia Millia Islamia, India
Shaobo He,
Central South University, China
Omar Abu Arqub,
Al-Balqa Applied University, Jordan

*CORRESPONDENCE
Vinod Patidar
vinod.patidar@spsu.ac.in;
patidar.vinod@gmail.com

SPECIALTY SECTION
This article was submitted to Dynamical Systems, a section of the journal Frontiers in Applied Mathematics and Statistics

RECEIVED 08 September 2022
ACCEPTED 04 October 2022
PUBLISHED 28 October 2022

CITATION
Kaur G, Agarwal R and Patidar V (2022) Image encryption using fractional integral transforms: Vulnerabilities, threats, and future scope. *Front. Appl. Math. Stat.* 8:1039758. doi: 10.3389/fams.2022.1039758

Image encryption using fractional integral transforms: Vulnerabilities, threats, and future scope

Gurpreet Kaur [1], Rekha Agarwal[1] and Vinod Patidar [2]*

[1]Amity Institute of Information Technology, Amity University, Noida, India, [2]Sir Padampat Singhania University, Udaipur, Rajasthan, India

With the enormous usage of digital media in almost every sphere from education to entertainment, the security of sensitive information has been a concern. As images are the most frequently used means to convey information, the issue related to the privacy preservation needs to be addressed in each of the application domains. There are various security methods proposed by researchers from time to time. This paper presents a review of various image encryption schemes based on fractional integral transform. As the fractional integral transforms have evolved through their applications from optical signal processing to digital signal and digital image processing over the decades. In this article, we have adopted an architecture and corresponding domain-based taxonomy to classify various existing schemes in the literature. The schemes are classified according to the implementation platform, that may be an optical setup comprising of the spatial modulators, lenses, and charge-coupled devices or it can be a mathematical modeling of such transforms. Various schemes are classified according to the methodology adopted in each of them and a comparative analysis is also presented in tabular form. Based on the observations, the work is converged into a summary of various challenges and some constructive guidelines are provided for consideration in future works. Such a narrative review of encryption algorithm based on various architectural schematics in fractional integral transforms has not been presented before at one place.

KEYWORDS

fractional integral transform, image encryption, double random phase encoding, discrete fractional Fourier transform, robust encryption

Introduction

Fractional transforms are the generalization of full transforms which we refer to as ordinary transforms in a more generic sense. Interestingly, the idea of fractional order in a transform first came into existence in 1695 during discussions between Leibnez and L' Hospital [1]: "Can the meaning of derivatives with integer order be generalized to derivatives with non-integer orders?" The question that was put up more than 300 years ago did not get a solution till the work on fractional calculus got explored. Later Jean-Baptiste Joseph Fourier in 1807 made important contributions to the study of trigonometric series and claimed that a periodic signal could be represented by a series of harmonically related sinusoids for the solution of 1D problems. Thus, the well-known Fourier transform is named in honor of Joseph Fourier for his significant contribution and application of the Fourier transform (FT) in many scientific disciplines. However, with the ever-expanding scope of research, it was found that FT has some shortcomings. As it is a holistic transform, the time domain signal is converted to the frequency domain and therefore is able to analyze only time-invariant signals. In other words, it is not possible to obtain a local time-frequency analysis which is pivotal for processing a time-variant or nonstationary signal. Thus, fractional Fourier transforms (FrFT), Short time Fourier transform (STFT), Wigner-Ville distribution, Wavelet transform, Gabor transform etc. were proposed as an alternative.

The initial work on fractional transform by Namias [2] presented a theory on fractional powers of Fourier transform and its application to quantum mechanics. The formal mathematical elaboration to Namias's theory was given by Mc Bride and Kerr [3]. Later, Lohmann [4] illustrated the relation of FrFT to Wigner rotation and image rotation. Almeida [5] further elaborated the concept by proposing a time-frequency representation of FrFT. Further, Ozakatas and Mendelovic proposed optical implementation and interpretation of FrFT [6–8]. With the evolution of digital channels, the digital computation of FrFT [9] and its discrete version [10] gave a new perspective to the application of FrFT in optical signal processing and related applications [11]. Pei et al. [12] established a relationship between FrFT and Discrete fractional Fourier transform (DFrFT) using Hermite eigen vectors based on the postulate in [13]. Various methods of DFrFT representations are given [14–16] with the extension to other similar transform domains [17–20]. We won't elaborate much on the mathematical details of the transforms here, interested readers may refer to above-mentioned references for the mathematical aspect of integral transforms and more specifically fractional Fourier transform and its variants. However, we give a conceptual description of the definition of fractional integral transforms. The term "fractional" in a transform indicates that some parameter has non-integer value. We can define any integral transform of the input function, $f(x)$ using any transform operator, as:

$$T\left[f\left(x\right)\right]\left(u\right)=\int_{-\infty}^{\infty}K\left(x,u\right)f\left(x\right)dx \tag{1}$$

where $K\left(x,u\right)$ is operator kernel. For example, in Fourier transform, $K\left(x,u\right) = \exp\left(-i2\pi ux\right)$. If it is a fractional transform then the operator is denoted as T^{α} with 'α' as a parameter of fractionalization. Therefore,

$$T^{\alpha}\left[f\left(x\right)\right]\left(u\right)=\int_{-\infty}^{\infty}K\left(\alpha,x,u\right)f\left(x\right)dx \tag{2}$$

For instance, continuous fractional Fourier transform is the generalization of a continuous Fourier transform. The ath order continuous fractional Fourier Transform of a function, $y(t)$, is given as:

$$Y_{\alpha}\left(u\right)=\int_{-\infty}^{+\infty}Q_{a}\left(u,t\right)y(t)dt \tag{2.a}$$

where $Q_{a}\left(u,t\right)$ is transform kernel given by

$$\begin{aligned}Q_{a}\left(u,t\right) &= \sqrt{1-jcot\alpha}.e^{j\pi\left(t^{2}cot\alpha-2tucsc(\alpha)+u^{2}cot\alpha\right)}\\ &= \sum_{k=0}^{\infty}\exp\left(-\frac{jk\alpha\pi}{2}\right)\psi_{k}\left(t\right).\psi_{k}\left(u\right)\end{aligned} \tag{2.b}$$

$\psi_{k}\left(t\right)$ *is kth*-order Hermite Gaussian function , $\alpha = a\pi/2$

$$\psi_{k}\left(t\right)=\frac{2^{\frac{1}{4}}}{\sqrt{2^{k}\,k!}}H_{k}\left(\sqrt{2\pi}\,t\right)e^{-\pi t^{2}} \tag{2.c}$$

where H_k is k^{th} Hermite polynomial with k real zeros.

For the discrete version of these fractional transforms, the postulate of discrete Fourier transform (DFT) is followed. As, $N \times N$ DFT matrix F is defined as

$$F_{kn}=\frac{1}{\sqrt{N}}e^{-\frac{j2\pi}{N}.kn}\;0\le k,n\le N-1 \tag{2.d}$$

where N is the length of the input sequence. Thus, αth order $N \times N$ DFRFT matrix is defined [12] as:

$$\begin{aligned}F^{\alpha} &= V\,\Lambda^{a}\,V^{T}\\ &= \begin{cases}\sum_{k=0}^{N-1}e^{-\frac{j\pi}{2}ka}v_{k}v_{k}^{T}, & for\ N:odd\\ \sum_{k=0}^{N-2}e^{-\frac{j\pi}{2}ka}v_{k}v_{k}^{T}+e^{-\frac{j\pi}{2}Na}v_{N}v_{N}^{T}, & for\ N:even\end{cases}\end{aligned} \tag{3}$$

where $V = \left[v_{1}\,v_{2}\,\ldots\,v_{N-2}\,v_{N-1}\right]$ for $N:odd$ and $V = \left[v_{1}\,v_{2}\,\ldots\,v_{N-2}\,v_{N}\right]$ for $N:even$, v_k is kth-order Hermite-gaussian like eigenvector, Λ is a diagonal matrix with its diagonal entries corresponding to eigenvalues of each column vector v_k. However, there are certain properties [2, 6, 7] that are desirable for fractional integral transform used in Eq. (2). Some of them are:

1. The fractional transform has to be continuous for any real value of the parameter, 'α'.
2. It should be additive: $T^{\alpha_1+\alpha_2} = T^{\alpha_1}T^{\alpha_2}$.
3. It should be reproducible for full transform if the parameter is replaced by integer values.
4. For $\alpha = 1$, it should give $T^1 = T$, a full transform.
5. For $\alpha = 0$, it should give $T^0 = I$, an identity matrix.
6. From the additivity property,

$$\int_{-\infty}^{\infty}K\left(\alpha_{1},x,u\right).K(\alpha_{2},y,u)du=K\left(\alpha_{1}+\alpha_{2},x,y\right) \tag{4}$$

TABLE 1 Various fractional integral transforms.

Frequently used	Less frequently used
Fractional Fourier Transforms [15, 21–46]	Fractional Riesz Transforms
Fractional Cosine Transform [18, 20, 41, 47–49]	Fractional F-Kravchuk Transform
Fractional Sine Transforms [18, 20]	Fractional Cauchy Transforms
Fractional Hartley Transforms [50–54]	Fractional Slant Transform
Fractional Mellin Transforms [55–59]	Fractional Stieltjes Transforms
Fractional Angular Transform [60–64]	Fractional Abel Transforms
Fractional Hadamard Transforms [19]	Fractional Sumudu Transforms
Fractional Gyrator Transform [65–71]	Fractional Brownian Transforms
Fractional Hilbert Transforms	Fractional Walsh Transforms
Fractional Affine Transforms	Fractional JigsawTransforms
Fractional Random Transforms	Fractional Kekre Transforms
Fractional Hankel Transforms	Fractional Schrodinger Transforms
Fractional Radon Transforms	Fractional Riemann Derivative
Fractional Wigner Distribution	Fractional Fokker-Plank Equation
Fractional DCT Transforms	Fractional Lagendre Transform
Fractional Hilbert Transforms	
Fractional Laplace Transforms	
Fractional S –Transform	
Fractional Wavelet Transforms [69, 72]	
Fractional Dual Tree Complex Wavelet Transform	
Fractional Haar Transforms	
Fractional Polar Harmonic Transform	

It is likely to mention here that the fractional parameter in a fractional Fourier transform refers to an angle of rotation (Wigner distribution) [4]. In some references, the fractional parameter is represented as $\alpha = a\pi/2$, where a: fractional number. If the angle of rotation, $\alpha = 0$, the transform is said to be in purely time domain. If $\alpha = 1$, it gives the transformation to the frequency domain whereas if the parameter is some fractional value then the transformation output results in a collective time-frequency domain. Table 1 lists some of the fractional transforms that are used in various applications of signal processing. Very few of them are used for image encryption applications due to certain properties that are required to be fulfilled for cryptographic applications.

Contributions and outline

The major contributions of this review article are summarized as:

- Information regarding the background and evolution of fractional integral transforms and their application in image encryption.
- Detailed taxonomy on various methods and corresponding architectural schematics for implementing these transforms in different domains.
- A brief overview and recent developments in optical transforms for image encryption with a tabulated description of recent review articles and various cryptanalytic strategies that are adopted to break the encryption.
- Review recent articles on the digital implementation of fractional integral transforms that have been merged with other domains/schemes for enhanced of security. Each of the classification is separately described and reviewed.
- The performance parameters adopted to evaluate an image encryption scheme are also summarized for reference in the comparative analysis of schemes.
- Based on the observations made in the review article, some issues are highlighted along with some viable solutions. A set of constructive guidelines are summarized that may be helpful to future researchers in designing a robust and highly sensitive encryption algorithm based on digital implementation of these fractional integral transforms.

The paper is further organized into five more sections. Section Taxonomy of fractional integral transforms provides the taxonomy along with a description of each classification and the review. Section Performance metrics for image encryption elaborates on the performance measures of encryption algorithms. Section Comparative analysis provides a comparative analysis of the results of some recently proposed articles. A summary on observations based on the literature review is included in Section Observations based on published literature. The review is concluded in Section Conclusion.

Taxonomy of fractional integral transforms

The fractional integral transforms have evolved through their applications from optical signal processing to digital signal and digital image processing over the decades. In this article, we have adopted an architecture and corresponding domain-based taxonomy to classify various existing schemes in the literature. The architecture can be broadly classified on the bases of the platforms used for implementation as shown in Figure 1. The platform can

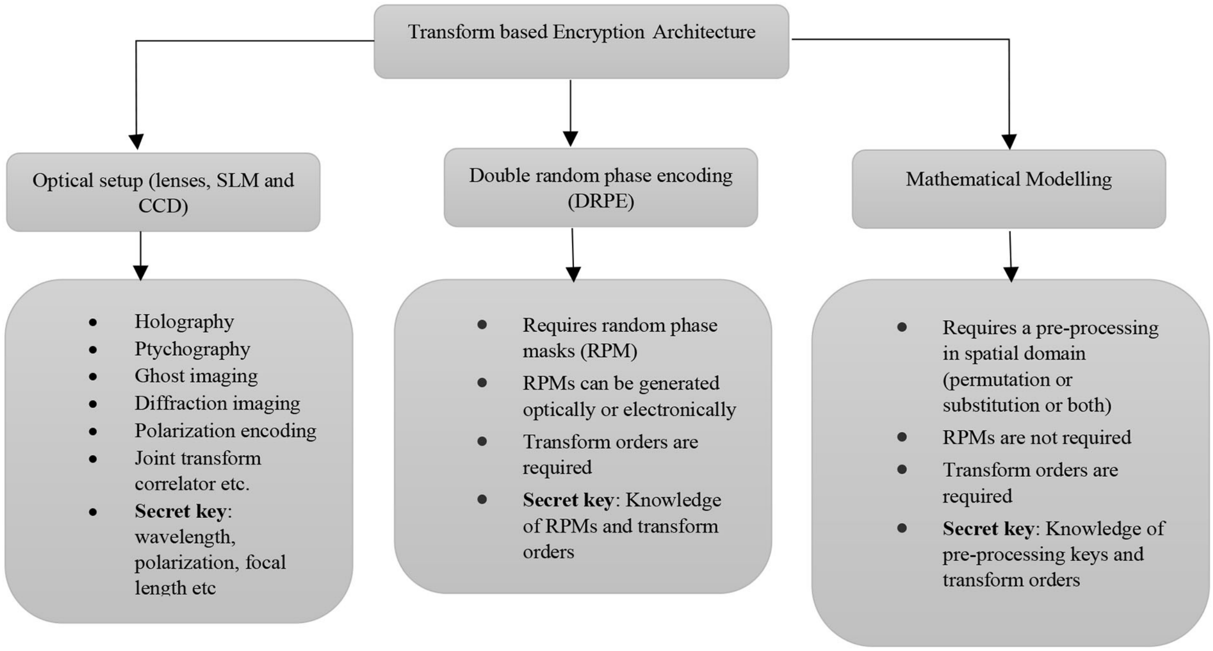

Transform based Encryption Architecture
Optical setup (lenses, SLM and CCD)
Double random phase encoding (DRPE)
Mathematical Modelling
• Holography
• Ptychography
• Ghost imaging
• Diffraction imaging
• Polarization encoding
• Joint transform correlator etc.
• Secret key: wavelength, polarization, focal length etc
• Requires random phase masks (RPM)
• RPMs can be generated optically or electronically
• Transform orders are required
• Secret key: Knowledge of RPMs and transform orders
• Requires a pre-processing in spatial domain (permutation or substitution or both)
• RPMs are not required
• Transform orders are required
• Secret key: Knowledge of pre-processing keys and transform orders

FIGURE 1
Classification of architectures for fractional transform-based image encryption.

be an optical setup that comprises of lenses, spatial light modulators (SLM), and charge-coupled devices (CCD). Another platform is based on the use of random phase masks (RPM) in transforming image pixels. Yet another is a digital platform, where mathematical modeling is followed to achieve the transformation.

Optical data processing

Optical data processing got introduced almost four decades before by Van der Lugt as an optical correlator which is based on the usage of the thin lens to produce two-dimensional Fourier transform of an image. This further led to the invention of other more advanced optical and optoelectronic processors. The classical methods for the realization of the optical scheme are based on two architectures [73]: a 4f-Vander Lugt (VL) and a joint transform correlator (JTC) architecture. In both of these methods, the input image is displayed in the form of transparency or as on SLM. With the advancement in technology, SLMs that are used these days are electrically addressed liquid crystal-based SLMs. The randomness in phase is obtained with ground glass or with a nonuniform coating of gelatine on glass plates. The RPMs thus obtained are recorded on SLMs during encryption or decryption. The outcome of a DRPE encryption is a random noise-like pattern with complex nature. In order to record these complex coefficients for storage and transmission, a holographic technique is required. Although both architectures require two RPMs to convert an image (amplitude or phase) to a stationary random noise, JTC is considered superior to VLC architecture. The VLC architecture requires conjugate RPMs and stringent alignment for decryption, whereas JTC does not require these two conditions and it is considered as alleviated from these limitations. Hence, a JTC architecture is considered superior to the VLC. To record the decrypted image, either a CCD (charge-couple device) or a conjugate of input plane RPM is used. In another method known as the optical phase conjugation method [74], a conjugation of an encrypted image is obtained with the use of optical phase conjugation in a photo-refractive crystal through 4 wave mixing. This phase conjugation can nullify the effect of RPM in the decryption process. A most recent classical implementation of fractional Fourier transform in terms of wave functions is presented in Weimann et al. [75].

We provide a brief overview of the various optical setups that are used for obtaining an optical transform of the scene or image. These are categorized as:

- Holographic methods: Holography is based on using an interference pattern generated by diffraction of the light field in 3 dimensions. Their resultant 3D image retains depth, parallax, and other such properties of the scene. Thus, the hologram is an unintelligible pattern formed by an image. Digital holography is further divided into two categories, namely, off-axis digital holography and phase-shifting digital holography. Javidi et al. [76] first presented a combined approach to providing image security through Double Random Phase encryption (DRPE) and holography. The author further extended his work to 3D information encryption [77]. Some of the most recent reviews are available in the literature [78, 79] that give insight into the evolution of this scheme over the last decade.

- Ptychography: It is based on coherent imaging generated using multiple probes that generate multiple diffraction patterns in a far field. Ptychography offers good quality of both recovered amplitude and phase distribution. Similar to holography, it also generates complex amplitude of the object but it does not require any reference beam like in holography. The application of Ptychography in image encryption has been proposed by many researchers [80–82] and most recently in [83, 84].
- Ghost imaging: It is also known as coherent imaging or two-photon imaging or photon-correlated imaging. It is a technique that produces an image formed by combining effects from two light detectors: one from the multipixel detector that does not view the object and another is a single pixel detector that views the object. Clemente et al. [85] proposed to use of ghost imaging for image encryption. Some of the recent works [86, 87] are based on a similar strategy.
- Diffractive imaging: It is referred to as imaging formed by a highly coherent beam of wavelike particles like electrons, X-rays, or other wavelike particles. The waves thus diffracted from the object form a pattern which is recorded on a detector. The pattern is used to reconstruct an image with an iterative feedback algorithm. The advantage of the absence of lenses is that the final image has no aberrations and therefore resolution is only dependent on the wavelength, aperture size, and exposure. The application of diffractive imaging in image encryption is proposed in Chen et al. [88], Quin et al. [89], He et al. [90] and Hazer et al. [91].
- Polarization encoding: An optical plane wave is used to illuminate the intensity key image and encoded into a polarization state. It is then passed through a polarizer (pixelated polarizer) to obtain the encrypted image. Gopinathan et al. [92] proposed to use of polarization encoding in image encryption. Some of the recent works in encryption application are proposed in Wang et al. [93].
- Joint Transform Correlators: The joint power spectrum of the plane image and key codes are the encrypted data in the JTCs [94]. Joint correlator-based encryption uses the same key code for decryption as used in encryption. This is unlike a classical DRPE scheme where a conjugate key is required. Many encryption schemes have been recently proposed based on JTC in fractional transform domain [65, 95].
- Phase retrieval method: In addition to the methods described above, there is an iterative phase retrieval method [96–98] wherein a digital approach is usually applied for embedding the input image into phase-only mask(POM), and either a digital or optical method is employed for image decryption. The main objective of a phase retrieval algorithm is to find either the correct or an estimate of POM under some constraint for a measured amplitude. Phase retrieval algorithms can be 2D or 3D. Unlike holographic-based or diffractive imaging-based optical encoding, a phase retrieval-based optical security system generates POMs as ciphertexts. Various transform domains such as FrFT and Gyrator transform can be employed in these encoding schemes.

Advantages of optical encryption

1. Optical instruments such as SLM and lenses have inherent characteristics of parallel processing.
2. Optical encryption methods possess multiple-dimensional and multiple-parameter capabilities. The optical parameters for security keys can be wavelength, polarization, and phase.
3. For optical encryption, researchers require multidisciplinary knowledge regarding optical signal processing, image processing, optical theories, and computer technologies as well.

Applications of optical signal processing

Fractional transforms and more precisely, fractional Fourier transform have gained keen interest from researchers in the area of optical signal processing. Thus, it is also commonly referred to as "Fourier Optics" or "Information optics." Fractional transforms have a widespread application in signal processing and image processing, in the area of time-variant signal filtering, phase retrieval, image restoration, pattern recognition, tomography, image compression, encryption, and watermarking. This article focuses on the image encryption application of various fractional integral transforms.

DRPE model for image encryption

DRPE-based image encryption has its roots in the work of Refregier and Javidi [21] where two random-phase functions in fractional Fourier domains are used to encrypt input plain image into stationary white noise. Hennelly and Sheridan [99] have shown image encryption as random shifting in the fractional Fourier domain. Unnikrishnan [22] has generalized the DRPE scheme in the fractional Fourier domain. The DRPE architecture is most exhaustively used and explored in various optical processing-based applications. The research community has been continuously exploring the possibilities to improve the security of DRPE [23, 50, 66, 67, 100] and has also successfully extended the DRPE scheme to other linear canonical transforms (LCTs) domains. Figure 2 shows the schematic architecture of DRPE-based image encryption scheme. As shown in Figure 2, there are two RPMs also known as POMs. One of the POM is placed at the input plane and another is placed at the Fourier plane. The POM$_1$ at the input plane makes the input signal/image white noise-like but nonstationary and POM$_2$ at the Fourier plane is also a white noise but is stationary. Let POM$_1$ at the input plane be $exp(j\phi\left(x,y\right))$ and POM$_2$ at Fourier plane as $exp(j\varphi\left(\mu,\nu\right))$, both being randomly distributed in the range $[0,2\pi]$. Therefore, wavefront after POM$_1$ is given by

$$F\left(\mu,\nu\right)=FT\left\{I\left(x,y\right)\exp\left(j\varphi\left(x,y\right)\right)\right\} \tag{5}$$

where $I\left(x,y\right)$ is input image in the spatial domain, FT denotes a Fourier transform operation. The wavefront, $F\left(\mu,\vartheta\right)$, gets modified by POM$_2$ in the Fourier domain and an inverse Fourier (IFT) is performed over it. This gives a complex domain wavefront as

$$C\left(\xi,\ \eta\right)=IFT\{F\left(\mu,\nu\right)\exp\left[j\phi\left(\mu,\nu\right)\right]\} \tag{6}$$

The complex-valued coefficients are recorded on a CCD in optical processing while the terms can be electronically recorded in a computer. During the decryption/reverse process, the complex domain wavefront is first transformed to POM$_2$ as

$$\hat{F}(\mu,\nu)=\{FT\left[\hat{C}\left(\xi,\ \eta\right)\right]\left\{\exp\left(j\phi\left(\mu,\nu\right)\right)\right\}^{*} \tag{7}$$

where $*$ represents a conjugate operation. IFT of Fourier wavefront is obtained with POM$_1$ conjugate as

$$\hat{I}\left(x,y\right)=\left\{IFT\left[\hat{F}\left(\mu,\nu\right)\right]\right\}\left\{\exp\left(j\varphi\left(x,y\right)\right)\right\}^{*} \tag{8}$$

Thus, $\hat{I}\left(x,y\right)$ is the decoded wavefront in the spatial domain.

DRPE schemes are broadly classified as (1) Amplitude-only DRPE where decoding is done without using POM$_1$. (2) Full-phase DRPE where the input image is fully converted into a full-phase map. This POM is used to encode images with the DRPE procedure. The only difference is that the input image is first normalized and converted into a phase image as $\exp\left[jI\left(x,y\right)\right]$ before encoding. Details of each classification are beyond the scope of this review work. However, it is likely to mention that each POM at the input as well as Fourier domain can be used as secret keys. This enlarges the key space thereby enhancing security.

Previous review articles and contributed evaluations

There are many review articles available in the literature [101–103] that provide the evolution of classical DRPE-based architecture. Some of the significant contributions in reviewing fractional transforms are listed in Table 2. The contribution of these reviews is summarized on various aspects and evaluations included in them. Each review article is categorized according to the evaluation of various schemes in the work. Whereas some of these are based on just conceptual and theoretical aspects, while others provide an evaluation of quantitative, qualitative, comparative, applications, etc. We have nomenclated these evaluations from E01 to E09 based on the criteria mentioned at the bottom of Table 2.

This will give better clarity to the reader and future researchers regarding various aspects discussed in each review. It is not possible to include all the related work in this paper for the sake of brevity. However, best efforts are put to include the most recent developments in DRPE-based encryption schemes as listed in Table 3. DRPE-based architecture has been extensively used and is considered as an effective method. DRPE methods require an RPM as the secret key that needs to be stored at the receiver for decryption. Besides that, a careful alignment of the RPM with received encrypted data has to be done. The inherent property of linearity and symmetricity proves to be a bane of encryption applications as the linearity may lead to vulnerability to different types of attacks. Based on these vulnerabilities, some of the recent works on cryptanalysis are summarized in Table 4. Each reference is included with a short description of the work and methodology adopted to cryptanalyze the security scheme.

Mathematical modeling of optical transforms with FRFT and its variants

LCTs, time-frequency transforms, and fractional Fourier transform (FrFT) are closely related. Since the application of FrFT to signal processing is proposed [4, 5, 8], there has been tremendous development in the application of FrFT and its variants to image encryption. As fractional transform orders serve as the secret key, the digital implementation is particularly suitable for encryption applications [99]. Since this work is mainly focused on the application of fractional transform in image encryption only, we won't elaborate the mathematical eloquence behind the fractional transforms here. This section specifically emphasizes the discrete realizations (DFrFT) and their application to image encryption. There are various methods proposed in the literature for the discretization of fractional transforms; some of them are classified as shown in Table 5 with pros and cons of each type. It is worth noting that Table 5 includes only a fractional version of Fourier transform. This is due to the fact that the fractionalization of LCTs started with Fourier transform itself and later was extended to other transform domains. The methods of discretization mentioned below are therefore conceptually applicable to variants of Fourier transforms as well, namely, Gyrator transform [57, 66],

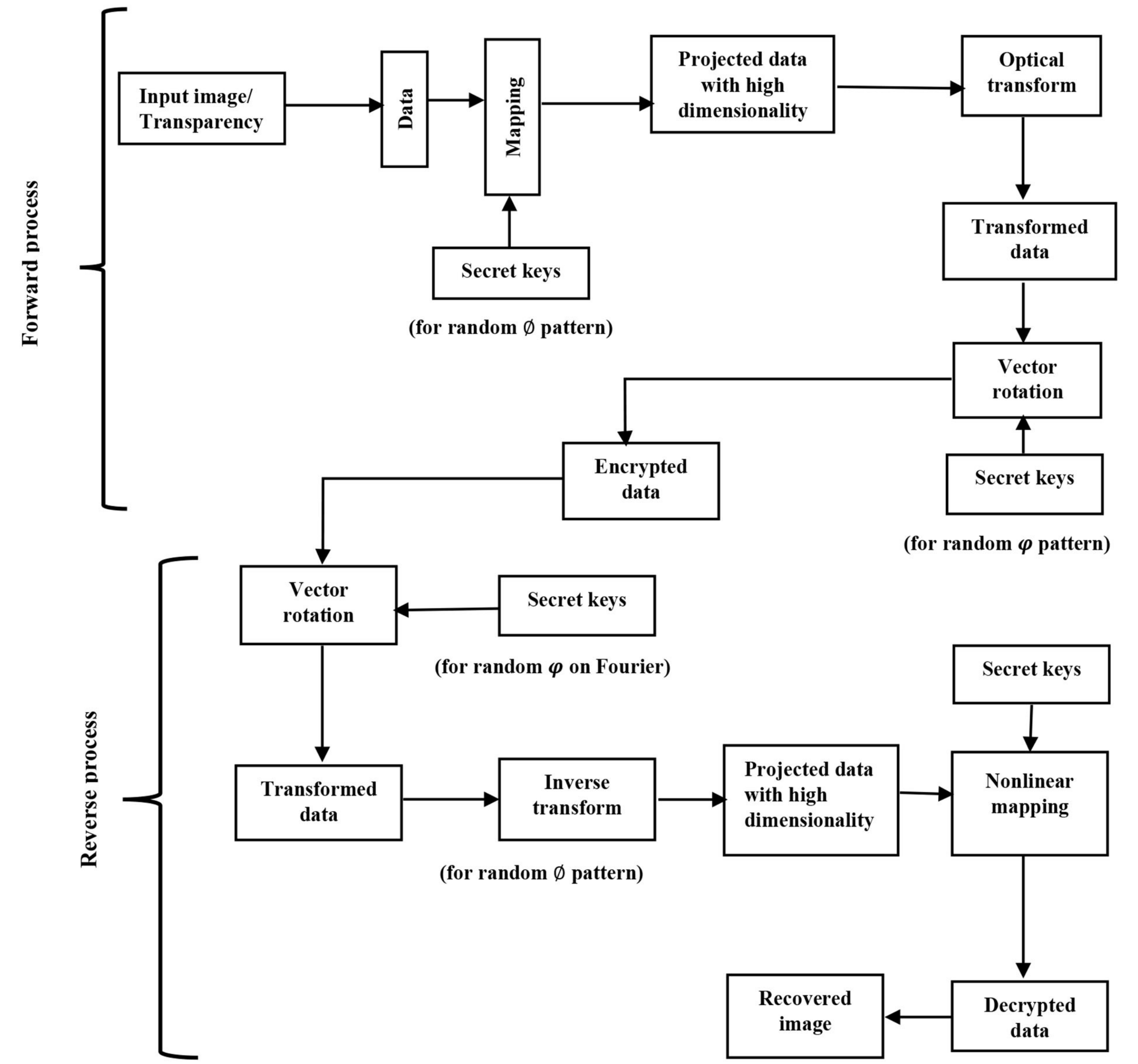


FIGURE 2
Architectural model for DRPE-based encryption scheme.

Mellin transform [25, 26, 58], Hillbert transform [137], Hartley transform [17, 20], Hadamard transform [19], etc.

Figure 3 shows the basic architecture for fractional transform-based image encryption that is digitally implemented without an RPM in either domain (without DRPE). As depicted in Figure 1, this method requires the knowledge of fractional transform orders that are used along both dimensions within a range [0,1]. The decryption is exactly similar to the forward process and requires the same fractional orders but with negative values to decrypt the image correctly. The encryption is thus a symmetric scheme and a slight change in the key value will result in incorrect decryption.

The major limitation of such a scheme is shorter key space which makes it vulnerable to brute force attacks. The input image is pre-processed for enhanced security and enlarging a key space. The pre-processing can be a scrambling operation that only shuffles the pixel positions to make the image, unintelligible. In some cases, this pre-processing can be a nonlinear operation that can be a substitution of pixel intensity values. There are various schemes that employ either scrambling [27–29, 47], substitution [30] or both [23, 48, 138] to enhance the security. The following section includes all major schemes that are proposed to improve the performance of fractional transform-based image encryption. We have categorized them

TABLE 2 Recent review articles on fractional transforms-based image encryption schemes.

Author[Ref]	Year	Description	Evaluations done
Moreno and Ferreira [101]	2010	On the usage of optical signal processing and its conceptual and theoretical details	E01, E08
Sejdić et al. [104]	2011	On FrFT digital realizations and related application areas	E01, E05, E06, E09
Saxena and Singh [105]	2013	On FrFT and its properties, versions in the discrete domain and some application areas	E01, E05, E09
Chen et al. [102]	2014	On the advances in optical security, various optical signal processing schemes illustrated	E01, E02, E06, E07, E08, E09
Yang et al. [106]	2016	On fractional calculus and MATLAB functions defined for same, various application areas reviewed	E01, E02, E05
Javidi et al. [103]	2016	On recent advances and challenges of optical security using free space optics, cryptanalysis and road map to the development of secure theory in optics.	E01, E02, E05, E06, E08, E09
Guo and Muniraj [107]	2016	On the vulnerability of LCT-DRPE based encryption to COA with numerical implementation	E01, E02, E03, E07, E08
Situ and Wang [108]	2017	A review on phase problems in optical imaging	E01, E05, E07, E08, E09
Guo et al. [97]	2017	On recent development in iterative phase retrieval and application in information security	E01, E02, E05, E07, E08, E09
Kaurl and Kumar [109]	2018	On the latest developments in the meta-heuristic methods of image encryption	E01, E02, E03, E04, E06, E07, E09
Jinming et al. [110]	2018	On research progress in theory and applications of fractional Fourier transform	E01, E02, E05, E06, E07
Gadhrili et al. [111]	2019	On different algorithms for color image encryption	E02, E03, E04
Jindal and Singh [112]	2019	On the applications of fractional transforms in image processing	E04, E07
Gómez-Echavarría et al. [113]	2020	On the applications of fractional Fourier transform in biomedical signal processing	E01, E05

E01, Conceptual and Theoretical; E02, Quantitative; E03, Qualitative; E04, Comparative on results; E05, Applications explored; E06, Vulnerabilities; E07, Architecture; E08, DRPE based; E09, Mathematical details.

in accordance with the strategical amalgamation of scheme with fractional transform domain. The schemes proposed in the literature are nomenclated in eight major categories (T01–T08). Each amalgamated scheme is reviewed separately. This portion of review article is elaborated as our emphasis is on the digital implementation of fractional integral transforms for image encryption.

Reality preserving with optical transform domain (T01)

The optical transform results in complex coefficients output corresponding to a real domain input image. Although it is easy to process these complex coefficients with a holography method but in a digital domain, it requires two images to be processed in the encrypted domain, one for real terms and other for imaginary terms. Therefore, storage and transmission increase complexity and overhead in digital channels. To overcome this limitation, Venturi and Duhamel [139] proposed a mathematical solution based on the properties of the complex transform output. Reality preserving refers to real domain output for a real domain input signal. The algorithm still has computational complexity, $O(N^2)$ for matrix order of N. Reality preserving transforms that are formulated with this algorithm have most of the required properties of fractional transforms along with a monotonously decreasing decorrelation power. Such transforms are beneficial where orthogonal reality preserving transform is required with their decorrelation power controlled by some parameters such as in joint source and channel coding. Initially, the algorithm was proposed in fractional sine and cosine transforms. It is further extended to other transforms with the basic properties of the transforms retained well. Recently, Zhao et. al [25, 59] used it to obtain fractional Mellin transform for triple image encryption. Reality preserving is also used in discrete fractional Cosine transform (FrCT) [47, 140], fractional Angular transform [60, 61], fractional Hartley transform [52–54, 141], besides fractional Fourier transform [28, 29, 31].

TABLE 3 Recent publications on evolutionary methods adopted in optical transform with DRPE-based architecture (2016–2021).

References	Method	Security	Advantages	Limitations
Abd-El-Atty et al. [114]	Based on the application of DRPE and quantum walks. An alternate quantum walk (AQW) is used to generate random masks as well as for permutation.	Moderate	1. Higher key space 2. Resistance to digital and quantum computer attacks.	1. Non uniform histograms 2. Classical attack analysis missing 3. Differential attack analysis not discussed.
Zhou et al. [115]	Image is transformed in DRPE domain. The phase information is quantized for its usage in the authentication. The plaintext is compressed by CS where the measurement matrix is also quantized using a sigmoid function.	High	1. Simultaneous compression and encryption. 2. Faster and efficient. 3. Robust to differential attacks	1. Higher complexity 2. PSNR is lower indicating degraded reconstructed image.
Huang et al. [116]	Low-frequency subbands are extracted by contourlet transform. Scrambled with 2D logistic map. 2DLCT is applied to obtain phase truncation and phase reservation. This is followed by an XOR operation with a logistic map.	High	1. Multiple image encryption 2. Uniform histograms 3. optimum entropy and CC of encrypted 4. Robust to classical and differential attacks	1. Performance degrades considerably with data loss and noise attack
Wang et al. [55]	Based on apertured Mellin transform realized by log-polar transform followed by apertured fractional Fourier transform.	High	1. Key size increased 2. Non linearity in transform is able to resist potential attacks	1. Quality of decrypted images vary with aperture length parameter 2. Mellin transform gives a lossy recovery, resulting in significant degradation in recovered image
Huang et al. [98]	Original image is encoded with a modified Gerchberg-Saxton algorithm, which is controlled by hyperchaos system derived from Chen chaotic map. Josephus traversing is used for scrambling the phase function followed by diffusion-confusion by hyperchaos.	High	1. Uniform histograms 2. High sensitivity to keys 3. Optimum entropy 4. Resistant to all potential attacks	1. Hyperchaotic map has high complexity in hardware implementation. 2. G-S algorithm based on hyperchaos increase encryption/decryption time
Huo et al. [117]	Based on DNA theory with DRPE technique with PWLCM based keys and random phase masks. Initial values of PWLCM are generated by massage digest algo5(MD5). Two rounds of process gives ciphertext.	High	1. High security to input keys 2. key space is large	1. Axis alignment is required for optical setup 2. Lack in differential attack analysis
Liansheng et al. [100]	Based on customized data container. Using phase masks that are generated from Hadamard matrix to collect intensities of data containers. After XOR coding, data is scrambled with logistic map	High	1. Solves issues related to inherent linearity of computation ghost imaging. 2. High sensitivity to keys	1D logistic map has its own limitations
Gong et al. [118]	Based on compressive sensing (CS) and public key RSA algo with optical compressive imaging system to sample input image. Walsh Hadamard transform, followed by scrambling with compound chaos	High	1. Enlarged key space 2. Resistant to CPA 3. Entropy is optimum for both global and local values 4. Robust to noise and data loss attack	1. Higher complexity for implementation
Chen et al. [119]	Chaotic Ushiki map is used to generate random phase masks. A single intensity image is encrypted from color image. An equal modulus decomposition used to create asymmetric keys	High	1. Enhanced security by Ushiki chaotic map 2. Enlarged key space 3. Immune to CPA and KPA	1. Lossy recovery 2. Entropy not reported 3. Differential attack analysis not done

(Continued)

TABLE 3 (Continued)

References	Method	Security	Advantages	Limitations
Yadav et al. [51]	Input is first transformed with chaotic Arnold transform. Phase masks are based on devil's vortex Fresnel lens (DVFL)	High	1. Use of DVFL eliminates axis-alignment issues. 2. Parameters of DVFL, orders of FrHT and AT serve as secret key	Robustness to classical and differential attacks not presented
Faragallah et al. [50]	Arnold transform is used to scramble RGB of image followed by a Fresnel based Hartley transform from random phase masks generated with a Logistic adjusted sine map	High	1. Enhanced security due to enlarged key size 2. limitations of logistic map are eliminated 3. Optimal CC of encrypted	1. Histograms are not independent of plane image input to some extent 2. UACI=0 3. Leakage of information due to low entropy values
Kumar et al. [120]	security key generated from a phase retrieval algorithm is used obtain 2D non-separable linear canonical transform of complex image formed by combining two plane images	High	1. Double image encryption with asymmetric keys 2. Robust to data loss attack 3. Chosen plain text attack addressed	1. Phase retrieval has its inherent complexity
Jiao et al. [121]	QR (quick response) code for speckle noise removal in Fresnel based optical transform	High	1. Speckle noise reduced in optical transformed output	1. Applicable only to gray scale images
Khurana et al. [122]	Phase-truncated Fourier and discrete cosine transform (PTFDCT) with random phase as keys. Decryption requires a cube root operation	High	1. Robust to differential attack 2. Enhanced security 3. Enlarged key space	1. Entropy is less than optimum 2. Correlation plots show unequal distributions along both dimensions leading to information leakage.
Su et al. [123]	Chaotic phase masks for cascaded Fresnel transform holography and constrained optimization for retrieval	Moderate	1. Reduces retrieval time using constrained optimization 2. Key sensitivity high due to use of chaotic Henon map	1. decrypted image is considerably deteriorated 2. performance will degrade under noisy and occlusion attacks
Li et al. [124]	Depth conversion integral imaging and hybrid cellular automata (CA)	High	1. PSNR of reconstructed images degraded with noise are higher 2. Key space is high (multidimensional) 3. Good resistance to data loss attack	1. Lossy decryption 2. Differential attack analysis not proved

Although certain probable drawbacks/limitations are mentioned corresponding to each scheme, some specific solutions like security enhancement methods can be applied in practice.

Application of chaos theory in optical transforms-based image encryption (T02)

Chaos theory refers to the study of unpredictable behavior in systems governed by deterministic laws. Chaotic properties are closely related to cryptography [142] owing to their sensitivity to initial conditions, randomness and ergodicity. Due to such intrinsic characteristics, chaotic maps have been extensively used in data encryption. Chaotic maps are used as pseudorandom generators [143], for substitution, and permutation of image pixels. Various schemes for encryption based on permutation only [144, 145], or substitution only [146] or a combination of both [138, 143] with the usage of either one-dimensional basic maps like logistic [147], sine, the tent [148], 2D Chirikov standard map [143], or higher dimensional compound chaos or higher dimensional hyperchaotic maps [149–151], depending on the application and level of security.

Chaotic maps have been extensively used in amalgamation with optical transforms-based image encryption for enhancing security. Fractional transform-based image encryption schemes have only transform orders as the secret key. However, this key space is not large enough and is therefore vulnerable to cryptanalysis. To enhance security, chaotic maps are used that also enlarge the key space. There are various schemes proposed in the literature that have used permutation with chaotic maps along with an optical transform [23, 28, 29, 50, 66, 69, 72, 152]. The order in which these two schemes are amalgamated may vary. Permutation in the spatial domain followed by transform or transform followed by permutation in the transform domain.

TABLE 4 Cryptanalytic approaches in optical/DRPE-based encryption schemes (2016–2021).

Author	Year	Description	Methodology/ strategy
Guo et al. [107]	2016	Phase retrieval attacks on LCT based DRPE schemes	Hybrid input–output algorithm, error reduction algorithm, and combinations of both type of phase retrieval algorithms are applied for ciphertext-only attacks on Separable LCT DRPE system.
Yuan et al. [125]	2016	Cryptanalysis and its remedy in encryption based on computational ghost imaging	Due to linear relation between input and output of the encryption with computational ghost imaging is attacked.
Li et al. [126]	2016	Vulnerability of impulse attack-free DRPE scheme to chosen plaintext attack	CPA on impulse attack free-DRPE is breached using a new three-dimensional phase retrieval algorithm.
Wang et al. [127]	2016	Cryptanalysis in phase space	Phase space information vulnerable to chosen plaintext attack (CPA) and known plain text attack (KPA).
Liao et al. [128]	2017	Ciphertext only attack on optical cryptosystem	Based on autocorrelation between plaintext and ciphertext, COA is imposed.
Hai et al. [129]	2018	Cryptanalysis of DRPE scheme with deep learning	Vulnerability to CPA with working mechanism-based learning with neural network.
Xiong et al. [130]	2018	Cryptanalysis of optical cryptosystem with combined phase truncated Fourier transform and nonlinear operations	A phase retrieval attack with normalization and bilateral filter is proposed.
Dou et al. [131]	2019	Known plaintext attack in JTC-DRPE scheme	Application of denoizing operations make the cryptosystem linear. Thus, KPA is possible.
Xiong et al. [24]	2019	Cryptanalysis in optical encryption based on vector decomposition of Fourier plane	Cascaded EMD (equal modulus decomposition)-based cryptosystem is attacked with CPA and a special attack.
Chang et al. [132]	2020	Ciphertext only attack in optical scanning cryptography (OSC)	A linear system property analyzed in the ciphertext expression equation of OSC lead to COA.
Jiao et al. [133]	2020	Known plaintext attack in cryptosystem based on space and polarization encoding	Matrix regression based on training samples is proposed to crack a space-based optical encoding and double random polarization encoding with KPA.
Zhou et al. [134]	2020	Vulnerability of encryption scheme based on diffractive imaging to machine learning attacks	An end-to-end machine-learning strategy is adopted to establish relationship between ciphertext and plaintext in case of diffractive imaging.
He et al. [135]	2020	Cryptanalysis of optical cryptosystem using untrained neural network	Untrained NN is used to break a phase-truncated Fourier transform-based optical asymmetric cryptosystem. Parameters are optimized by plain-ciphertext encryption model of phase truncated Fourier transform.
Song et al. [136]	2021	Cryptanalysis of phase only information as it is vulnerable to chosen plaintext attack.	Deep learning structure is trained using sparse phase information of the encrypted domain image as phase only information is vulnerable to classical attacks.
Li et al. [126]	2016	Vulnerability of impulse attack-free DRPE scheme to chosen plaintext attack	CPA on impulse attack free-DRPE is breached using a new three-dimensional phase retrieval algorithm.
Wang et al. [127]	2016	Cryptanalysis in phase space	Phase space information vulnerable to chosen plaintext attack (CPA) and known plain text attack (KPA).
Liao et al. [128]	2017	Ciphertext only attack on optical cryptosystem	Based on autocorrelation between plaintext and ciphertext, COA is imposed.
Hai et al. [129]	2018	Cryptanalysis of DRPE scheme with deep learning	Vulnerability to CPA with working mechanism-based learning with neural network.

Some of the schemes follow substitution-permutation and transform collectively [138, 153, 154] to further enhance security. We have reviewed some of the most recently proposed schemes that use chaos-based permutation/substitution with optical transforms.

Wu et al. [48] proposed a color image encryption scheme in random fractional discrete cosine transform (RFrDCT) along with scrambling and diffusion paradigm (DSD). A logistic map is used to generate a randomized vector of fractional order. This enlarges key space and increases sensitivity.

TABLE 5 Various methods for discretization of Linear Canonical transforms.

Type	References	Pros	Cons
Sampling type DFrFT	[68]	A direct and simplest of all methods	Discrete version is derived at the cost of losing many important properties like unitary, reversibility, and additivity. Therefore, it has limited applications.
Improved Sampling type DFrFT	[9]	It works like a continuous FrFT and is a fast algo	Doesn't have orthogonal and additive property. Also, it requires to put some constraints on input signal.
Eigen vector decomposition based DFrFT	[10, 12, 16, 17]	Based on eigen values and eigen vector of DFT matrix and then evaluating their fractional power. Retains orthogonality, reversibility, and additivity. Further improved by orthogonal projection in [12]	This type of DFrFT lack fast computation, and the eigen vectors cannot be written in closed form.
Linear combination type DFrFT	[13, 19, 20]	Eigen vectors are derived by linear combination of identity operation, DFT, time inverse operation and IDFT. Satisfies properties of reversibility, additivity and orthogonality.	The outcome of transform does not match with continuous transform. It works very much similar to Fourier transform and lose characteristics of fractionalization of powers.
Chirp type DFrFT	[56]	DFrFT is derived as multiplication of DFT and periodic chirp signals. Satisfies additivity, reversibility property along with Wigner distribution's rotation property.	There are constraints on the selection of rotation angles and also N (sample length) should not be a prime number. This makes it complicated
Closed form DFrFT	[15]	Derived 2 types of DFrFT and Discrete Affine transform (DAFT). Performance is similar to continuous FrFT for Type I and can be calculated using FFT. Type II is improved form of Type I and is applicable to signal processing. Has lowest complexity.	Scaling property exists for only Type I and not for Type II.

A multiple parameter fractional Hartley transform (FrHT) is proposed by Kang et al. [141] with its reality preserved for a color image encryption. The chaos is embedded into the algorithm at each step. The original color image with individual color components is first combined into a single image. This single image is divided into different sub-blocks. The blocks are then shuffled based on a pseudo-random sequence generated from non-adjacent-coupled map lattices (NCML) based on logistic maps. The initial parameters of NCML are generated from yet another chaotic map (Arnold Cat map). The initial parameters of chaotic maps at this stage serves as secret keys. Next stage of encryption is based on a pixel scrambling operator which is based on a 2D Chirikov standard chaotic map (CSM). Using CSM, a series of 2D and 3D angle matrices are generated that are used to convert images in RGB space to newer space. The final stage is to obtain an MPFrHT in real domain (RPMPFrHT) and to divide the image into three to get concatenated encrypted image as ciphertext.

A new fractional transform coined as the non-separable fractional Fourier transform is proposed by Ran et al. [32]. RPMs are generated by Arnold transform. The advantage of this type of transform is that it is able to tangle information along and across two dimensions together. It is closely related to the Gyrator transform. Also, the proposed scheme is resistant to decryption with multiple keys, unlike ordinary fractional Fourier transform.

Wu et al. [155] proposed a RFrDCT for image encryption. The RFrDCT domain image is subjected to confusion-diffusion paradigm. The confusion is obtained using a game-of-life (GoL) algorithm and diffusion in the next stage is based on an *XOR* operation with another chaotic map. The initial parameters of chaos serve as secret keys of encryption. Enhanced performance is claimed with the adopted strategy. A perturbation factor is applied for resistance against differential attacks.

An encryption scheme with S-box generation is proposed in Wu et al. [72] which is unique in the way these S-boxes are generated. Chaotic Chebyshev map and linear fractional transform are used for the construction of S-box. Partial image encryption is achieved by a permutation-substitution-diffusion (PSD) network and multiple chaotic maps in the linear wavelet transform (LWT) domain. Using dynamic keys for controlling encryption aids in security against differential attacks. Partial encryption of only sensitive portions not only reduces computation complexity but is also faster and more efficient.

Jamal et al. [156] proposed yet another scheme that uses a combination of linear fractional transform and chaotic systems to generate substitution boxes for image

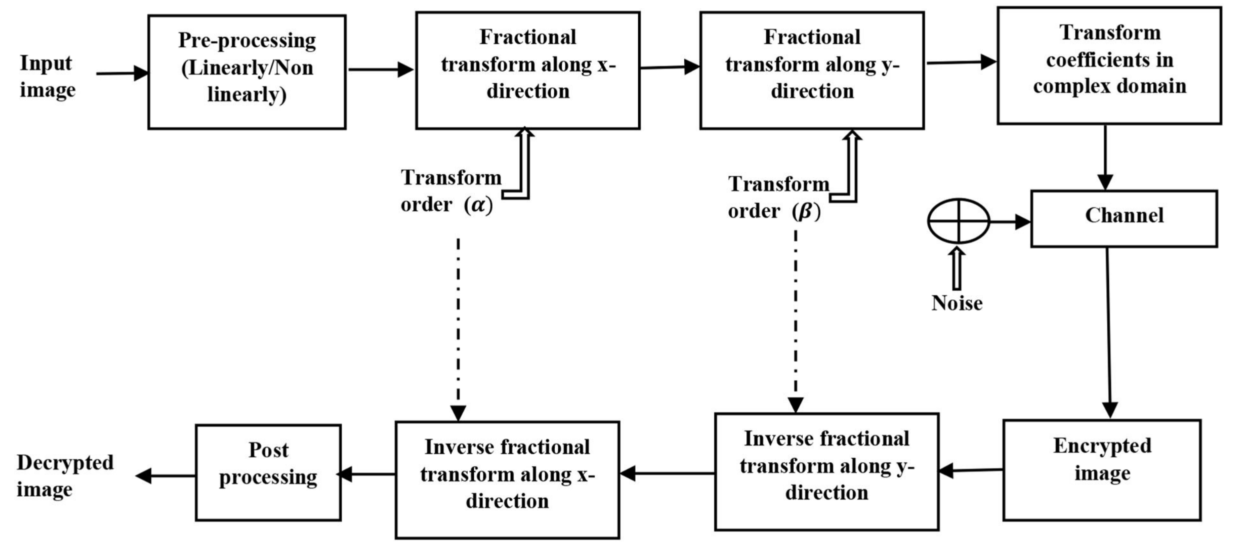

Input image
Pre-processing (Linearly/Non linearly)
Fractional transform along x-direction
Fractional transform along y-direction
Transform coefficients in complex domain
Transform order (α)
Transform order (β)
Channel
Noise
Decrypted image
Post processing
Inverse fractional transform along x-direction
Inverse fractional transform along y-direction
Encrypted image

FIGURE 3
Schematic architecture for Fractional transform-based image encryption in digital domain.

encryption. The chaotic maps used in the scheme are generated from a combination of seed maps to enhance the security and chaotic range. The investigation for complexity thus obtained with the proposed scheme is based on various algebraic and statistical tests. The investigation gives testimony of improved perplexity and confusion in the encrypted domain.

A novel Fresnel-based Hartley transform is proposed in Faragallah [50] for an optical-double color image encryption scheme. The color image is first separated into individual channels and are scrambled separately with the Arnold transform (AT) in spatial domain. Each scrambled image is then multiplied with a 2D chaotic Sine-adjusted logistic map (LASM) and then a Hartley transform is applied to each channel. This procedure is repeated once again with another set of AT-based scrambling (now in Hartley domain), and then each channel is multiplied with another set of 2D-LASM. The final step is obtaining inverse Hartley transform which gives an outcome across each channel in Fresnel domain. The color channels in Fresnel domain are concatenated to obtain a single image which is the final ciphered image.

A fractional angular transform (FrAT) is used in Sui et al. [62] where plain image is substituted with a chaotic logistic map prior to transform. The transform orders along with initial value of logistic map serve as secret keys of encryption. The scheme performs marginally as there are certain limitations due to similarity in histograms of plain and encrypted domain and correlation coefficients in encrypted domain are considerably higher. Moreover, the scheme is not evaluated for entropy measure and differential attack analysis.

Compressive sensing (T03)

Compressive sensing (CS), also referred to sparse signal sampling, was introduced by work of Donoho, Candes [157, 158]. CS is able to achieve compression and signal sampling simultaneously [118, 159, 160]. For a signal of bandwidth, $BW = \Omega$, the sampling frequency (f_s) required to represent the signal is much smaller than Nyquist frequency ($f_s \ll \Omega$). Let R^N be the set of N-tuples of real numbers. If $x \in R^N$ is input 1D signal sampled using CS, then x can be sparsely represented using an appropriate basis function $\Psi = [\psi_1, \psi_2 \ldots \psi_N]$. Thus, $x = \Psi_s = \sum_{i=1}^{N} s_i \psi_i$. Let $y_{M\times N}$ be the measured matrix with $M \ll N$. Then, $y = \varnothing x = \varnothing \Psi_s = As$ where $y \in R^N$. Thus if measurement matrix, A that is used to measure sparse signal, s is given, then the construction of signal requires solving an underdetermined linear system and the sparse signal can be obtained by solving a combinatorial optimization problem given by : $\min \|s\|_0 \, : y = \varnothing \ \Psi_s = A \ s$.

A collective compression-encryption scheme is proposed in Santhanam and McClellan [26] with 2D compressive sensing and fractional Mellin transform. The original image is first measured using a measurement matrix in both dimensions to reduce data volume with 2D CS. The measurement matrix is constructed using partial Hadamard matrices. Chaos is used to control the measurement matrix with its initial conditions. The non-linear Mellin transform is used to overcome the security issue related to linear transform.

Zhao et al. [161] proposed a double-image encryption scheme which is claimed to be faster and more efficient. The scheme utilizes DWT as the basis for the measurement matrix. Both images are first transformed into DWT basis and are compressed with the measurement matrix derived from 2D Sine-Logistic modulation map (2D-SLMM). The images are then combined and Arnold transformation is applied for scrambling the coefficients. Two circular random matrices are generated using 2D-SLMM with different seed values. These random matrices are used to obtain DFrRT. The encrypted image is thus in DFrRT domain.

In another CS-based scheme proposed by Zhang et al. [33], Kronecker product (KP) is combined with the chaotic map for the generation of measurement matrix and RPMs. Low-dimensionality seed maps are extended to high-dimensional KP. These high-dimensional maps are used for the measurement matrix. The scheme is able to provide an efficient and fast approach to color image encryption.

A comparatively simpler scheme is proposed in Deng et al. [162] where image compression-encryption uses a combination of 2D CS and DFrRT. The basis function for the measurement matrix is a discrete cosine transform (DCT). The measurement matrix is constructed with a chaotic logistic map to control row vectors of the Hadamard matrix. The compressed image is then encrypted by DFrRT. Reconstruction of CS requires Newton's smoothed l_0 norm (NSL_0) algorithm.

An asymmetric cryptosystem for color images based on CS and equal modulus decomposition (EMD) is proposed by Chen et al. [163]. In this scheme, the color image is initially combined to a single image. With the application of DWT, this image is converted into low-frequency and high-frequency images. The high-frequency image is compressed by a measurement matrix generated from logistic map. The compressed image is segmented into two matrices. One of the matrices is used as a private key (a random matrix related to the plain image) for DFrRT and another matrix is combined with the low-frequency image to form a complex function. This complex function is transformed into DFrRT with the private key (random matrix) that is plain image-dependent. This enables the cryptosystem to resist known and chosen plaintext attacks. The output of DFrRT is decomposed into 2 masks using EMD where one mask is a cipher image and another is a private key. The inverse CS in the decryption process is based on the basis pursuit (BP) algorithm.

Yi et al. [34] proposed to use multiple measurement matrices instead of a single measurement matrix that is used to sample all blocks of an image. This strategy enables to overcome the issue of chosen plaintext attacks. The mother measurement matrix is derived from a single chaotic map and other measurement matrices are generated by exchanging rows using a random row exchanging method. However, another chaotic map is required to control the row-exchanging operation. The compressed image is then transformed with FrFT. The transform is followed by two consecutive pixel scrambling operations to guarantee nonlinearity and to increase key sensitivity in the proposed scheme. Ye et al. [164] proposed a compressed-sensed color image encryption scheme based on quaternion discrete multi-fractional random transform with the hash function SHA-512. The parameters of chaos are updated by randomly selected hash values. The use of multifunctional transform not only increases the key space but also improves the key sensitivity.

On the basis of fixed/multiparameter (T04)

Fractional transforms can decorrelate the spatial domain pixels based on the fractional value of the transform orders. The fractional transforms are also looked upon as Wigner distribution where each fractional order corresponds to an angle of rotation in the optical domain [4]. With a fixed value of transform orders, the key space is limited and the cryptosystem is vulnerable to brute force attack. To overcome this limitation, various researchers proposed to use multiple parameter-based fractional transforms [35–39, 153, 165] with their own definitions and postulates. Mathematically, a FrFT has multiplicity which is due to different choices of both Eigen function and eigen value classes [35]. Thus, the multiplicity is intrinsic in a fractional operator. Lang [31] proposed a multiparameter FrFT where the periodicity of M is utilized. The transform order vector, n, can be M-dimensional integer vector. This provides an extra degree of freedom as the periodicity parameter; M serves as a secret key along with the vector parameters.

Sui et al. [63] proposed a multiparameter discrete fractional angular transform (MPFAT) for image encryption that uses fractional order and periodicity parameters to provide multiple parameters in the transform. Similar to a discrete fractional Angular transform (DFAT), MPDFAT also satisfies properties such as linearity, multiplicity, and index additivity. Zhong et al. [166] proposed a discrete multiple parameter FrFT (DMPFrFT) for image encryption using the periodicity parameter for extending to multiple parameters.

Azoug et al. [23] proposed yet another opto-digital image encryption with a multiple parameter DFrFT after a non-linear pre-processing of the image in spatial domain with a chaotic map. The multiparameter scheme is extended based on the work of Pei et al. [40] which extend the DFrFT to have multiple order parameters equal to the number of input data points. If all the parameters are made equal in an MPDFrFT, then it reduces to a single parameter DFrFT.

A general theoretical framework of MPDFrFT is presented in Kang et al. [153]. The work proposed two different frameworks as Type I and Type II MPDFrFT that include existing multiparameter transforms as their special cases. Further, an in-detail analysis of the properties of such transforms is discussed and higher dimensional operators are also defined. Some new types of transforms such as MPDFrCT, MPDFrST, and MPDFrHT (Cosine, Sine, Hartley) are constructed under the proposed framework along with their applications such as feature extraction and 2D image encryption.

A quaternion algebra is used with multiple parameter fractional Fourier transform (MPFrQFT) by Chen et al. [30]

for generalizing MPFrFT. Both forward and reverse MPFrQFT transform are defined and a color image encryption based on the proposed transform is evaluated for its performance as compared to other encryption algorithms. The proposed scheme has larger key space and is more sensitive to transform orders.

Ren et al. [41] proposed a multiple image encryption scheme based on discrete multiple parameter fractional Fourier transform (DMPFrFT) for which original images are filtered in DCT domain and multiplexed into a single image. The multiple parameters are again generated using a periodicity parameter which serves as one of the keys. Other keys are the parameters for scrambling the multiplexed image (random matrix), and transform orders of DMPFrFT.

A multiparameter discrete fractional Hartley transforms for image encryption is proposed by Kang and Tao [141]. The multiple parameters are generated by extending the fractional order to N-dimensional vector and the FRHT kernel is represented as a linear summation with weighting coefficients.

DNA sequence (T05)

DNA coding method is inferred from the Deoxyribonucleic acid and is a branch of computing based on DNA, biochemistry and molecular biology hardware. DNA sequences appear in the form of double helices in living cells. A DNA code is simply a code of alphabetic set $Q = \{A, T, C, G\}$. These alphabets refer to 4 nucleic acid bases: A (adenine), C (cytosine), G (guanine), and T (thymine): A and T, G and C are complimentary. The complimentary rules are referred to as Watson-Crick compliment [167]**.** Thus, pairing can be described as: $A = T$, $T = A$, $C = G$, $G = C$ and if a binary code is given to each as $00, 11, 01, 10$ with $(00, 11)$ and $(01, 10)$ as complimentary. With vector algebraic operations based on DNA computing [168, 169]**,** pixel permutation and substitution can be performed if the image pixels are represented in the form of binary sequences.

Recently Farah et. al [27] proposed to use FRFT along with chaos and DNA for image encryption. Initially, a random phase matrix is generated using a chaotic Lorenz map. The plain image is converted to a binary matrix and encoded according to chosen DNA encoding rule. Also, the random phase matrix is encoded to DNA sequence with the same rule. The coded plain image is *XOR*ed with that of the encoded random phase matrix. Using the RPMs generated from the 3D chaotic map (Lorenz map), iterative FrFT is performed and the resultant image is XORed with the third chaotic sequence to obtain the final ciphered image.

An optical image encryption set-up based on DNA coding is proposed by Huo et al. [117] where a piecewise linear chaotic map (PWLCM) is used to generate a key matrix as well as a random phase matrix. A message digest hash algorithm (MD5) is used to generate initial values of PWLCM. An MD5 hash of plaintext consists of 128 bits. *XOR* operation for DNA is used. Initially, the plain image and key matrix are converted to binary sequences with DNA coding rules that are different for different rows in the image. The DNA-encoded plain image is *XOR*ed with a key matrix and a forward Fresnel domain DRPE is applied to obtain the final-ciphered image.

Cellular automata (T06)

Cellular Automata (CA) also called cellular spaces, tessellation automata/structures, cellular structures, or iteration arrays find application in various fields like physics, microstructure modeling etc. CA consists of regular rigid cells that are generated in accordance with a fixed rule which is nothing but a mathematical function. CA is used in cryptography due to the possibility of pseudo-random number generation with such rule (Rule 30) which is a class III rule displaying aperiodic chaotic behavior [42, 170]. Li et. al [171] proposed a 3D image encryption using computer-generated integral imaging (CIIR) and cellular automata transform. An elemental image array (EIA) recorded by light rays coming from 3D image is mapped according to a ray-tracing theory. An encrypted image is then generated from 2D EIA using cellular automata transform. It is claimed that CA-based encryption is error-free and being an orthogonal transformation, it offers simplicity. The performance of the scheme is measured in terms of bit correct ratio (BCR) and PSNR for reconstructed and is compared to some similar proposed schemes. This scheme of combining optical transforms to that of CA is unique in its methodology. Recently, there is no further exploration of the proposed idea.

Double image (T07.1)/multiple image (T07.2)

Double image encryption schemes are aimed to provide more efficiency in terms of resources. A double image is simultaneously encrypted and decrypted. Such schemes also provide higher speed and better sensitivity besides less storage space requirement. Therefore, double image encryption schemes have drawn attention of various researchers [29, 63, 70, 152, 161, 172].

Recently, Yuan et al. [173] proposed an image authentication with double image encryption based on non-separable fractional Fourier transform (NFrFT). The two images are combined to form a complex image matrix and is transformed with NFrFT. The output of the transform is also a complex matrix. The transform orders and coefficient parameters serve as secret keys. Novelty of the proposed work is in the selection of a partial phase that is reserved for decryption. A nonlinear correlation algorithm is to authenticate the two recovered images. The cross-correlation of two compared images is referred to as non-linear correlation (NC) whose strength is specified by a parameter, $k \in [0, 1]$. An appropriate value of k is selected to authenticate the images. Peak to correlation energy (PCE) is a ratio of maximum peak intensity value and total energy of the non-linear correlation plane. Thus, PCE is measured to determine k and hence authenticity.

A double image encryption scheme based on interference and logistic map is proposed in Liansheng et al. [174] to overcome the silhouette problem. The two input images are initially joined to make an enlarged image. This joined image is subjected to scrambling based on chaotic sequence generated from a logistic map. Then, the scrambled image is again separated into two. One of the images is directly used to generate two-phase keys/masks based on optical interference. Another scrambled image is encrypted with DRPE method using first phase mask (key). This is followed by multiplying the complex outcome with another phase mask for transformation to the ciphertext. The author suggests to use input parameters of the logistic map, wavelength and axial mask as secret encryption keys to further enhance the security.

Singh et al. [67] proposed a full-phase encryption scheme for its better security compared to amplitude image. The scheme uses two spatial domain input images and converts each of them to a phase image. The phase images are then multiplied with RPMs and transformed in the Gyrator domain with rotation angle, α. The gyrator domain images are then added and subtracted to get two intermediate images. The intermediate images are then bonded with structured phase masks based on the Devils vortex lens (DVFL) specified with certain parameters. This is followed by another Gyrator transform with a different rotation angle, β to obtain two encrypted images. Decryption is exactly the inverse of the encryption process.

Similar to double image encryption schemes, there is another category where multiple images are simultaneously encrypted to reduce the key space as compared to the data to be encrypted (images) but at the cost of increased complexity [69, 175]. Recently Sui et al. [64] proposed a double image encryption where two images are initially combined into a single image along the column of the first image followed by the second image. This combined image is scrambled with a 2D sine logistic modulation map. Next, the scrambled image is divided into two components to constitute a complex image. One of the components is the phase part and another part is the amplitude of the complex image. The complex image is shared using Shamir's three-pass protocol where the encryption function is a multiparameter fractional angular transform which is preferred for its commutative property.

Sui et al. [43] proposed multiple image encryption with asymmetric keys in the FrFT domain. Initially, a sequence of chaotic pairs is generated using symmetrically coupled logistic maps. This chaotic sequence is used to scramble the spatial domain images. Phase only function (POF) of image is retrieved using an iterative process of FrFT domain. In the next stage, all the POFs are modulated into an interim which is transformed to real-value ciphertext by FrFT and chaotic diffusion. The three random phase functions are used as keys to retrieve POFs of plain images and three decryption keys are generated in the encryption process.

A multiple image encryption scheme is proposed [49] by combining a non-linear fractional Mellin transform with a FrCT. Fractional Mellin transform is used for its robustness to classical attacks. The original images are simultaneously transformed into a DCT domain and then re-encrypted with amplitude and phase encoding. The transformed images have changed center-coordinates due to fractional Mellin transform since FrMT is a log-polar transform of the image followed by a FrFT of log-polar image. The fractional orders of FrFT, phases ψ_j, θ_j are the secret keys.

Recently, Guleria et al. [176] proposed to encrypt three RGB images simultaneously using RSA cryptosystem followed by a discrete reality preserving FrCT and the final stage of scrambling with Arnold transform. To accomplish multiple image encryption, 3 RGB images are combined into a single image using a single color component of each image as R,G,B components. All three indexed images are individually ciphered with the proposed algorithm and then combined as a single ciphered image. The security of the scheme depends not only on the input parameters of RSA, Arnold transform and orders of transform but also on their sequence of arrangement. Decryption is exactly the inverse of the encryption scheme.

Watermarking in the encrypted domain (T08)

Recently, many researchers have proposed to use of optical transform for watermarking applications [69, 71, 177–179]. Watermarking an image is a data-hiding method for copyright protection and copy prevention. Depending on the application, a watermark can be a visible pattern or can be hidden in the host image. For copyright, its generally a visible pattern and for resolving an authorship problem, the watermark is secretly embedded into image which can be recovered by an authorized user only. In the latter case, the watermark is usually a binary logo that is encrypted into a noise-like pattern and then embedded in the image for enhanced security. Many researchers have followed this approach in the watermarking algorithm. Some of the recent watermarking schemes with an encryption algorithm using fractional transforms are reviewed in this section.

Singh et al. [180] proposed to embed an encrypted watermark in fractional Mellin transform (FrMT) into the host image. The two deterministic phase masks (DPM) are generated to be used in the input and frequency plane. The watermark image is first converted into a log-polar image. After multiplying the log-polar image with the first DPM, it is transformed to a FrFT domain. This is FrMT transformation. In the next step, again the second DPM is multiplied by the complex outcome and inverse FrFT is obtained. For embedding, the outcome is attenuated by a factor and then added to the host image. SVD decomposition is applied in the last stage to make the watermarked image unrecognizable and is transmitted as individual S, V, D matrices.

A quaternion algebra is used to define a quaternion discrete fractional random transform (QDFRNT) which generalizes DFRNT for its application in watermarking [181]. The host image is divided into blocks and QDFRNT is applied to each block. The scrambled watermark image is used to modify the mid-frequency coefficients of the QDFRNT host image. The transform orders and parameters of the scrambling scheme in the watermark image are used as secret keys of encryption.

Liu et al. [182] proposed a novel transform, known as fractional Krawchouk transform (FrKT), to generalize the Krawchouk transform. Derivation of FrKT is based on eigenvalue decomposition and eigen vectors. For validating the imperceptibility of the proposed transform, a watermarking application is illustrated in the work. A better robustness and imperceptibility with proposed transform have been claimed in the work.

Performance metrics for image encryption

Image data have high redundancy and large volumes as compared to text or binary data. It may also have some real-time operations or may also be incorporated with compressed data of a certain format. Thus, an image encryption scheme needs to satisfy certain requirements. Some of the commonly used performance requirements are discussed in this section. The categorization of such performance analysis is shown in Figure 4. Performance analysis of encryption requires a comprehensive investigation of perceptual security and cryptographic security. Perceptual analysis requires that the outcome of an algorithm is unintelligible to human perception whereas cryptographic analysis refers to the ability of the algorithm to resist cryptanalysis that includes all possible attacks in terms of the secret key, data statistics etc.

Perceptual security analysis

Perceptual security can be investigated with some subjective metrics [183]. The ciphertext can be classified into typical quality levels as shown in Table 6. QL0: signifies a completely recognizable image which indicates that the encryption is not valid, QL1: signifies a partially recognizable image contour like edges and boundaries are visible but the texture is not clear. QL2: signifies that the image is completely unintelligible and is considered perceptually secure.

Another measure of perceptual quality is done by evaluating a set of parameters for comparison of encrypted images with reference to the plain image. Some of the commonly used objective metrics are explained below.

i. *Peak signal to noise ratio* (*PSNR*): PSNR is the measure of spectral information in an image. A higher value indicates greater similarity in the test images. In an encryption algorithm, PSNR values are evaluated to quantify the dissimilarity in the encrypted image with respect to plain image. During decryption, the same measure indicates the efficacy of the algorithm in the reverse process. Practically *PSNR* $\geq$ 28 indicates that the test images are similar. For any pair of images, plain image (*P*) and ciphered image (*C*), the *PSNR* is mathematically defined as:

$$PSNR\,(P,C) = 10\log_{10}\frac{(L-1)^2}{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[P_{i,j}-C_{i,j}\right]^2} \tag{9}$$

ii. *Mean square error* (*MSE*): It is also an error metric like PSNR that indicates the dissimilarity between the test images. In an ideal case, for two similar images, *MSE* should be zero. *PSNR* and *MSE* are mathematically related to each other as:

$$PSNR\,(P,C) = 10\log_{10}\frac{(L-1)^2}{MSE} \tag{10}$$

$$\therefore\ MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[P_{i,j}-C_{i,j}]^2 \tag{11}$$

iii. *Spectral Distortion measure (SD)*: It indicates the spectral dissimilarity between the reference image and test image. The SD measure evaluates as to how far is the spectrum of the test image from that of the reference image. The spectral distortion is defined as:

$$SD\,(P,C) = \frac{1}{MN}\sum_{u=1}^{M}\sum_{v=1}^{N}|F_P\,(u,v) - F_C\,(u,v)| \tag{12}$$

where $F_P\,(u,v)$, $F_C\,(u,v)$ are Fourier transforms of plain image, $f_P(m,n)$ and encrypted image, $f_C(m,n)$, respectively.

iv. *Structural Similarity Index Measure (SSIM)*: Wang et al. [184] proposed a metric based on the human visual system (HVS) that considers biological factors, namely, luminance, contrast, and structural comparison between the image and a reference image. This measure known as *SSIM*, is used to quantify the visual image quality.

$$SSIM\left(x,y\right) = f\left(l\left(x,y\right),c\left(x,y\right),s\left(x,y\right)\ \right) \tag{13}$$

where *l(x,y), c(x,y) and s(x,y)* are luminance, contrast, and structural comparison, respectively. For any two pairs of images *P* and *C*, it is mathematically defined as:

$$SSIM\,(P,C) = \frac{(2\mu_P\mu_C +\ C_1)\,(2\sigma_{PC}+C_2)}{\left(\mu_P^2+\mu_C^2+C_1\right)\left(\sigma_P^2+\sigma_C^2+C_2\right)}\backslash n \tag{14}$$

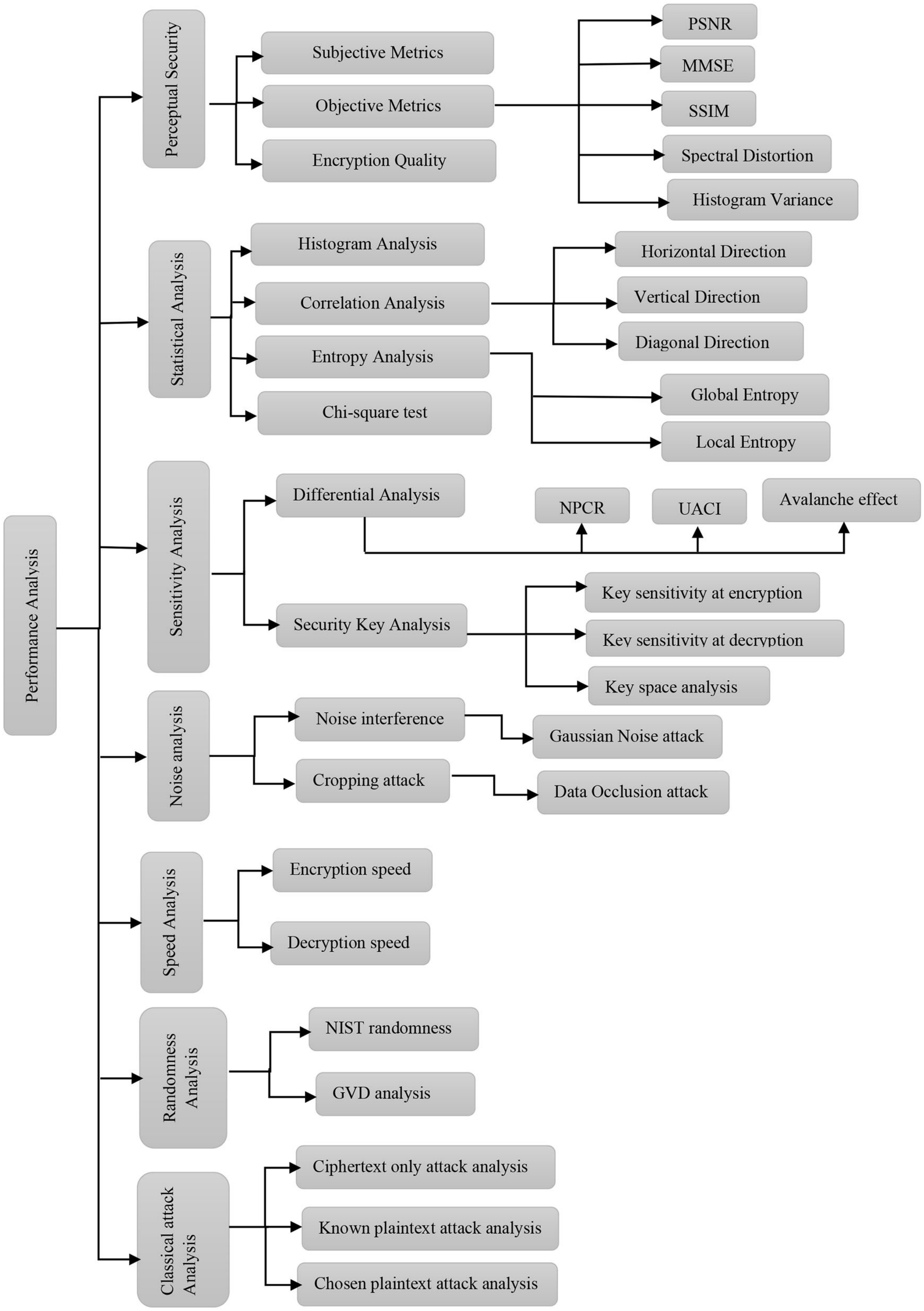


FIGURE 4
Performance requirements of image encryption scheme.

TABLE 6 Subjective metrics for perceptual security analysis.

Quality Level	Ciphertext quality
QL0	Image contours are completely recognizable
QL1	Partially recognizable contours of the image
QL2	Completely unintelligent/ white noise like image

v. *Histogram variance*: In order to quantify the uniformity of cipher images, variances of histograms are evaluated [185]. Variances are also evaluated for two different cipher images that are encrypted from two different secret keys on the same plain images. The lower values of variance indicate higher uniformity. The variance of histogram is mathematically evaluated as:

$$var\left(Z\right) = 1/n^2 \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{2}(z_i - z_j)^2 \qquad (15)$$

where $Z = \{z_1, z_2, z_3,, \ldots z_{256}\}$ is vector of histogram values, z_i, z_j are the number of pixels that have grey values equal to i and j, respectively.

vi. *Encryption Quality* is a subjective measure that collectively evaluates an algorithm for the level of security it provides. There are 4 different levels for evaluation as explained in Table 7.

Statistical analysis

According to Shannon's communication theory of perfect secrecy [186], "It is possible to evaluate most of the encryption techniques by statistical analysis". He suggested two methods for such analysis. One is histogram analysis and another is correlation analysis for the adjacent pixels in the encrypted image.

Histogram analysis

Histogram is the pixel frequency distribution where each grey level is plotted for the number of pixels with that particular value in the image. An effective cryptosystem should be able to generate ciphertext with fairly uniform histograms, which are also significantly different from the plaintext.

TABLE 7 Evaluation of encryption quality.

Security Level	Performance
SL0	High cryptography security + High perceptual equality (QL2)
SL1	High cryptography security +Low perceptual security (QL0, QL1)
SL2	Low cryptography security + High perceptual security (QL2)
SL3	Low cryptography security + Low perceptual security (QL0, QL1)

Chi-square test

In order to verify the uniformity of the histogram, a chi-square test is performed [187] and defined as:

$$\chi^2_{test} = \sum_{k=1}^{K}\frac{(o_i - e_i)^2}{e_i} \qquad (16)$$

where k is gray-level (256 for 8-bit image), o_i, e_i are the observed and expected times occurrence of each gray-level, respectively. The test is performed with different significance levels (generally at 0.05) for a null hypothesis.

Correlation analysis

For a perceptually meaningful image, the correlation between adjacent pixels is very high. It is necessary for an effective cryptosystem to significantly reduce these correlation values by decorrelating them in the encrypted domain. For such analysis, either all or a few pixels are randomly selected and correlation plots are obtained for horizontally, vertically, and diagonally adjacent pixels. The correlation plots in each direction should display the pixels to be uniformly scattered over the entire intensity range. For quantitative analysis, correlation coefficients are evaluated for two adjacent pixels in horizontal, vertical, and diagonal directions using Eqs. (17)–(19). For x_i, y_i as gray values of ith pair of selected adjacent pixels,

$$\rho_{(x,y)} = \frac{cov\left(x, y\right)}{\sqrt{D(x)}\sqrt{D\left(y\right)}} \qquad (17)$$

where $cov\left(x, y\right) = E[x - E\left(x\right))(y - E\left(y\right))]$

$$= \frac{1}{N}\sum_{i=1}^{i=N}[(x_i - \frac{1}{N}\sum_{i=1}^{i=N}x_i)^*(y_i - \frac{1}{N}\sum_{i=1}^{i=N}y_i)] \qquad (18)$$

$$D\left(x\right) = \frac{1}{N}\sum_{i=1}^{i=N}(x_i - \frac{1}{N}\sum_{i=1}^{i=N}x_i)^2,$$

$$D\left(y\right) = \frac{1}{N}\sum_{i=1}^{i=N}(y_i - \frac{1}{N}\sum_{i=1}^{i=N}y_i)^2 \qquad (19)$$

Entropy analysis

Information entropy is a mathematical property that depicts the randomness associated with the information source. The entropy of a message source s is given as:

$$H\left(d\right) = -\sum_{i=0}^{L-1} P\left(s_i\right)\log_2 P\left(s_i\right) \tag{20}$$

where L is the highest intensity value of pixels in image, s_i is the *ith* symbol in message, P(.) refers to the probability. The entropy defined in Eq. (20) is termed as Shannon's entropy [186]. Besides, a local entropy has been recently proposed [188] as an extension of Shannon's entropy measure. It is the mean entropy of several randomly selected non-overlapping blocks of information source. For an 8-bit image, $L = 256$, there are $K = 30$, nonoverlapping blocks to be randomly selected from the image with each block having 1,936 pixels (T_B=1936). Therefore, this entropy measure is also termed as (K,T_B)-local entropy and is evaluated using Eq. (21)

$$\overline{H_{k,T_B}}\left(S\right) = \sum_{i=1}^{k}\frac{H\left(S_i\right)}{k} \tag{21}$$

where S_i are randomly selected non-overlapping image blocks with T_B pixels in each block of S with total of L intensity scales.

Sensitivity analysis

Key sensitivity analysis

The sensitivity of an encryption scheme can be evaluated in two aspects: (1) at encryption stage which means that a completely different ciphertext should be generated with a very minute change in the input key value, (2) at the decryption stage, the ciphertext should not be correctly recovered if there is very slight change in the correct key values. Key sensitivity (KS) is mathematically defined as:

$$KS = \frac{1}{M\times N}\sum_{m=1}^{M}\sum_{n=1}^{N} C_1(m,n)\bigotimes C_2(m,n)\times\ 100\% \tag{22}$$

where C_1 and C_2 are two different ciphered images with slight change in key values corresponding to same plain image, P. $M \times N$ is total number of image pixels in the image.

$$C_1\left(m,n\right)\bigotimes C_2\left(m,n\right) = \begin{cases}1, C_1\left(m,n\right)\neq C_2\left(m,n\right)\\ 0, C_1\left(m,n\right) = C_2\left(m,n\right)\end{cases} \tag{23}$$

The value of KS should be as close to 100% [183].

Key space analysis

Key space refers to the set of all possible keys that are used in encryption of information. A brute force attack is possible if an intruder manages to make an exhaustive search on the set of possibilities until the correct one is found. Thus, feasibility of brute-force attack depends on the total number of valid keys. This number is an important feature to determine the strength of a cryptosystem, and it has to be large enough ($> 2^{100}$) [142] as per today's computing power.

Differential analysis

With reference to plaintext, the sensitivity refers to change in ciphertext with slight change in plaintext. This is termed as differential analysis where an adversary can change a single pixel in plaintext and compare the corresponding ciphertexts to get some clue about secret keys. The diffusion property of a cryptosystem enables it to spread any change in plaintext to the entire ciphertext. There are two indicators for numerical evaluation of resistance to such attack: NPCR (number of pixel change rate) and UACI (unified average change in intensity). Theoretically, the closer values of *NPCR* and *UACI* are 99.6093 and 33.4635%, respectively, indicating the effectiveness of the applied algorithm [189]. These indicators are mathematically defined as:

$$NPCR = \frac{1}{M\times N}\sum_{i,j} D\left(i,j\right)\times 100\% \tag{24}$$

$$UACI = \frac{1}{M\times N}\sum_{i,j}\frac{\left|C\left(i,j\right)-\tilde{C}\ \left(i,j\right)\right|}{L-1}\times 100\% \tag{25}$$

where C , $\tilde{C}$ are two encrypted images with the same keys but with a slight change in the corresponding plain image of size, [M N] with the highest intensity value, L.

$$D\left(i,j\right) = \begin{cases}1,\ C\left(i,j\right)\neq\tilde{C}\ \left(i,j\right)\\ \quad 0,\ otherwise\end{cases} \tag{26}$$

Avalanche effect

The avalanche criterion is referred to as an average number of bits that differ between C and $\tilde{C}$ while changing a pixel in plaintext. The ideal value of the avalanche effect is 0.5 (50%).

Noise analysis

The communication channels over which the image information is transferred are responsible for the addition of some noise in the form of degradation or distortion. The performance of a cryptosystem in such a scenario requires analysis. Gaussian noise with zero mean and varying values for variance is added to the encrypted image for *Gaussian noise analysis*. The quality of the decrypted image is checked in perceptual as well as numerical terms with different variances in noise [60, 190]. The results thus obtained are compared for the noise analysis. The *Occlusion attack* refers to the loss of data or cropping of a portion of the image due to noisy channels. The cryptosystem should be capable of recovering the appropriate amount of information even after some occlusion in data. In order to check for the robustness to occlusion attack, some pixels of encrypted image (10, 15, 25, 50, 75%) are cropped and corresponding decrypted image quality is evaluated in perceptual and numerical analysis [25, 66, 190].

Speed analysis

Speed analysis refers to the critical execution time for forward and reverse process in an encryption scheme. As typical configuration and capacity of a system greatly determine its computation speed, therefore a comparison of encryption and decryption time is a trivial task. Different machines perform differently. However, time analysis is an important feature, especially where real-time application is involved. Time analysis is performed in terms of encryption time and decryption time separately. Generally, a large sample set of images are considered for evaluating the average time taken in the encryption and decryption process on a present-day commonly used system configuration.

Randomness analysis

NIST SP800-22 is a statistical test suite for random and pseudorandom number generators that are used for cryptographic applications. The advantage of this test suite is that it does not require any assumptions on the generator. Rather, it only looks for a particular statistical recurrence in the generated sequence (random). It consists of 15 *p*-value-based tests that include frequency test, run test, and spectral test. These tests are generally not used in transform-based cryptography. However, we mention it here due to usage of it in some classical methods of image encryption.

GVD analysis

The gray value difference of a pixel form its four neighboring pixels in an image is given by:

$$G\left(i,j\right) = \sum \frac{\left[\, I\left(i,j\right) - I\left(i^{'},j^{'}\right)\right]}{4} \tag{27}$$

The average difference in gray values corresponding to each pixel in image is

$$G_{av}\left(i,j\right) = \frac{1}{(M-2)(N-2)} \sum_{i=2}^{M-1} \sum_{j=2}^{N-1} G\left(i,j\right) \tag{28}$$

Thus, gray value difference (GVD) parameter [191] of an encryption scheme is defined as:

$$GVD = \frac{G_{av}^{P}(i,j) - G_{av}^{C}(i,j)}{G_{av}^{P}\left(i,j\right) + G_{av}^{C}(i,j)} \tag{29}$$

where G_{av}^{P} and G_{av}^{C} are the average differences in gray values for original plain image and ciphered image, respectively. The ideal value of GVD parameter is unity. For a good encryption scheme, this parameter should be as close to 1.

Classical attack analysis

In cryptography, classical attacks are launched to cryptanalyze an encryption scheme. The adversary can have certain information regarding plain text or ciphertext that provide for cryptanalysis. If the adversary has access to set of ciphertext, then it can launch a *ciphertext only attack*. If it is able to get access to set of plain texts and corresponding ciphertexts, then a *known plaintext attack* can be launched. In a *chosen plaintext attack,* it is assumed that the adversary has access to arbitrary plaintexts and can obtain the corresponding ciphertexts. From the above-stated assumptions, a chosen plaintext attack provides the most information to the adversary. Thus, if a cryptosystem is able to resist chosen plaintext attack, it is believed to be able to resist other classical attacks as well [154, 192]. Therefore, an image encryption scheme should have excellent diffusion properties for providing robustness to a chosen plaintext attack analysis.

Comparative analysis

As shown in Table 8, each of the proposed schemes is accompanied by the parameters used to evaluate the encryption algorithm and the technique that is merged with the fractional transform. We have categorized these techniques into eight, as reality preserving (T01), chaos theory based (T02), compressive sensing (T03), multiple parameters (T04), DNA sequence (T05), cellular automata (T06), double image encryption (T07.1), multiple image encryption (T07.2), and with watermarking (T08). The comparative analysis is based on the results available for *Lena* image only. Table 9 illustrates the subjective comparison for the same references as listed in Table 8 along with the probable vulnerabilities associated with each of them. These vulnerabilities are expressed as V01–V07 (mentioned below the Table 9). It is worth mentioning here that the vulnerabilities of each scheme can be removed by specific methodology in practice.

It is evident from the values in Table 8 that studies in which chaos-based permutation or substitution is merged with fractional transform domain have higher entropy measure, low

TABLE 8 Comparative analysis for performance metrics of proposed schemes (for *Lena* image).

[Reference]	Year	Technique used	Correlation analysis			Average entropy	Key space	Average NPCR(%)	Average UACI(%)	Encryption quality
			Horizontal	Vertical	Diagonal					
Kaur et al. [53]	2021	T02, T05	0.0015	0.0014	0.0059	7.9952	10^{247}	99.6348	33.5816	SL0
Ye et al. [164]	2021	T02, T03,T04	–	–	–	–	2^{259}	–	–	SL1
Kaur et al. [54]	2021	T02, T05	0.0033	−0.0099	−0.0046	7.9768	10^{228}	99.5956	33.8798	SL1
Farah et al. [27]	2020	T02, T05	0.0693	0.0610	−0.0242	7.9991	—	99.5677	33.4353	SL0
Guleria et al. [176]	2020	T02,T07.2	0.0223	0.0187	0.0137	1.0149	10^{70}	99.4664	34.1316	SL1
Kaur and Agarwal [190]	2020	T01,T02	−0.0006	−0.0057	0.0009	7.9938	10^{102}	99.6006	34.6379	SL0
Kaur et al. [52]	2019	T01,T02, T07.2	0.0036	−0.0038	0.0023	7.99	–	–	–	SL2
Faragallah [50]	2018	T02,T07.1	0.0001	−0.0029	−0.0019	7.5907	–	99.7400	0	SL2
Zhang et al. [33]	2018	T02, T03	0.0127	0.0101	0.0139	–	10^{136}	–	–	SL2
Kang and Tao [141]	2018	T01, T02, T04	−0.0001	−0.0014	0.0004	–	–	99.8640	33.3330	SL0
Kang et al. [60]	2018	T01, T02, T04	0.0015	0.0017	−0.0033	–	$10^{98} = 2^{325}$	99.9949	33.3616	SL0
Mishra et al. [28]	2018	T02	0.0020	−0.0007	0.00006	7.4739	–	–	–	SL0
Ref. [29]	2018	T01, T02, T07.1	–	–	–					SL3
Kaur et al. [48]	2017	T02	0.01513	−0.0024	−0.0045	7.9974	2^{297}	–	–	SL2
Yu et al. [62]	2017	T02	0.1068	0.0766	0.0182	–	$\approx 10^{16}$	–	–	SL3
Deng et al. [162]	2017	T02, T03	0.0909	0.2389	0.0126	–	10^{37}	–	–	SL2
Pan et al. [49]	2017	T07.2	0.0249	0.0505	0.0280	–	$27^5 \times 30^5$	99.6279	33.4599	SL2
Sui et al. [64]	2016	T02, T07.1	–	–	–	–	10^{55}	–	–	SL2
Santhanam and McClellan [26]	2015	T02	0.0104	0.0299	0.0062	–	$10^{34} \times 13^5 \times 11^5$	–	–	SL2
Zhou et al. [161]	2015	T02, T03	0.0119	0.0925	0.0325	–	10^{64}	–	–	SL2
Singh et al. [67]	2015	T07.1	0.0093	0.0172	0.0021	–	–	–	–	SL2
Sui et al. [43]	2014	T07.2	0.0040	−0.0018	0.0266	7.9976	–	–	–	SL2

correlation coefficients, high NPCR and UACI, higher key space, excellent key sensitivity, robustness to noise and data occlusion attacks, hence having higher security levels. Reality preserving algorithm has contributed toward the digital implementation of optical transforms and has enabled researchers to overcome major limitations regarding complexity issues of fractional transforms in the digital domain. Compressive sensing is used to reduce the data deluge while dealing with large images for encryption but their performance is marginal in terms of higher correlation coefficients and vulnerability to leakage in information.

CS-based encryption schemes are highly complex [193] and reconstruction is time-consuming. It has been observed in the results of the above-reviewed articles that CS-based schemes lack uniform histograms in the encrypted domain and CC values are considerably higher. Also, CS-based simultaneous compression and encryption schemes are vulnerable to cryptanalysis due to linearity [194]. In a broad sense, if the plaintext is sparse, the key of the cryptosystem may not be safe as it is possible to exploit the prior sparsity knowledge to extract information of the key from ciphertext. The key and the plaintext may be partly accessed using some information processing technology such as Blind source separation (BSS) [195].

Multiple parameter-based fractional transform schemes perform better than fixed/single transform order-based schemes. This is due to enlarged key space and better uniformity in encrypted histograms. However, there are some deficiencies related to multiple parameter schemes [44–46] due to linearity that need to be avoided. The linear relation among consecutive transform orders and periodicity is the major limitation that can lead to multiple decryption keys corresponding to an encryption key. This depicts its vulnerability to various attacks. To overcome this issue, it is necessary to introduce some means of breaking the linear relationship among consecutive transform orders or by careful selection of transform orders through a random selection scheme [38], [190].

DNA sequence operation is little less explored with optical transforms. However, it is able to enhance security with increased key space and randomness in encrypted data. Double and multiple image encryption schemes are preferred for

TABLE 9 Comparative analysis for subjective parameters (refer Table 8 for performance metrics).

Reference	Metrics for perceptual analysis	Noise analysis	Occlusion attack	Classical attacks	Differential attack	Statistical attack	Time analysis	Probable Vulnerabilities
Kaur et al. [53]	✓	✓	✓	✓	✓	✓	✓	V06, V07
Ye et al. [164]	✓	✓	✓	X	X	✓	X	V02, V05, V07
Kaur et al. [54]	✓	✓	✓	✓	✓	✓	✓	V07
Farah et al. [27]	X	X	X	X	✓	✓	X	V01, V03, V06, V07
Guleria et al. [176]	✓	✓	✓	✓	✓	✓	X	V03, V07
Kaur and Agarwal [190]	✓	✓	✓	✓	✓	✓	X	V03, V07
Kaur et al. [52]	✓	X	X	X	X	✓	X	V01, V02,
Faragallah [50]	X	✓	✓	✓	✓	✓	✓	V02
Zhang et al. [33]	X	X	X	✓	X	✓	X	V01, V02, V07
Kang and Tao [141]	✓	✓	✓	✓	✓	✓	X	V03, V07
Kang et al. [60]	✓	✓	✓	✓	✓	✓	X	V07
Mishra et al. [28]	✓	X	X	X	X	✓	X	V03, V04, V05, V07
Kaur et al. [29]	✓	X	X	X	X	✓	X	V02, V04, V06, V07
Wu et al. [48]	✓	X	X	X	X	✓	X	V02, V03, V07
Yu et al.. [62]	X	✓	✓	X	X	✓	X	V02, V03, V07
Deng et al. [162]	✓	✓	✓	X	X	✓	X	V01, V02, V07
Pam et al. [49]	X	✓	✓	X	✓	✓	✓	V01
Sui et al. [64]	X	✓	✓	X	X	✓	X	V02, V03, V07
Santhanam and McClellan [26]	X	✓	✓	✓	X	✓	X	V02, V07
Zhou et al. [161]	✓	✓	✓	X	X	✓	✓	V02, V06
Singh et al. [67]	✓	✓	✓	X	X	✓	X	V02, V07
Sui et al. [43]	✓	✓	✓	✓	X	✓	X	V02, V07

V01, High complexity; V02, Low encryption quality; V03, Dependent on diffusion; V04, Smaller key space; V05, poor efficiency; V06, Lossy; V07, may not be applicable for real time applications.

speed and increasing encryption efficiency. Watermarking is another domain where fractional transforms are used to encrypt the watermark before embedding in a blind watermark scenario. The encryption of the watermark logo in the collective time-frequency domain increases the robustness to various attacks.

Observations based on published literature

In an exhaustive search performed in the month of December 2021 on the various online databases: ACM Digital library, Elsevier, Google Scholar, IEEE explore, Springer link, Taylor and Francis and Wiley for the number of research papers published related to the encryption of different multimedia contents during the period 2015–2021. The pictorial view to highlight the percentage of papers published on the encryption of various multimedia contents like: images, video, audio, text data etc. has been shown in Figure 5.

According to search results, it is observed that the number of publications is majorly in text and image encryption. However, the number of image encryption works is dominating with 42% of all the metadata available. We believe that it is due to the wide application area of image data, from platforms like social media to sensitive data like military and telemedicine fields. Almost every sector of communication is dependent on image transmission in one way or the other. It is also observed that amongst various mathematical implementations of the fractional transforms, FrFT is most popular with more than 60% of the total publications in fractional integral-based image encryption schemes. This is followed by fractional wavelet transform (FrWT) with a contribution of 16%, fractional Hartley transform, FrHT (10%), fractional Cosine transform, FrCT (7%) and the remaining few on other transforms (namely, Mellin, angular, sine etc.).

As the present manuscript is mainly concerned with image encryption using optical/fractional integral transforms, therefore, we narrowed down our search for the number of papers published year-wise on the fractional transform-based

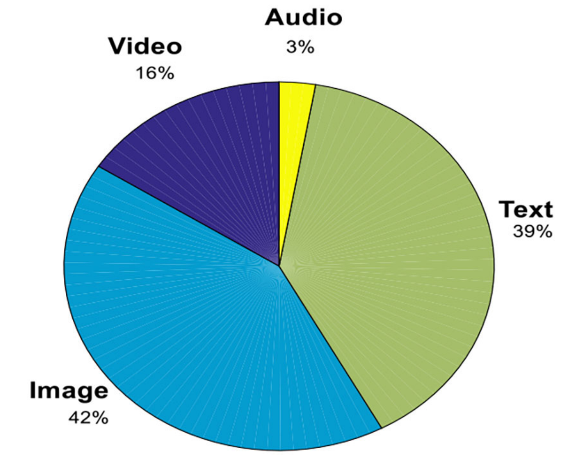


FIGURE 5
Percentage of research papers published on the encryption of various multimedia contents like: images, video, audio, text data during 2015–2021.

image encryption schemes. Figure 6 illustrates a graphical representation of the related publications in all the major online databases during the period 2015–2021.

It is observed that the number of publications on image encryption in the fractional transform domain has considerably increased every year. This gives testimony to the fact that with the advent of evolutionary algorithms based on fractional integral transforms in the digital domain has increased its popularity and is receiving significant attention from the researcher community.

It has been also observed that most of the encryption algorithms with fractional transform as the main component are evaluated for statistical analysis, noise attack, and occlusion attack analysis only. This is probably the reason for less popularity of optical transform-based image encryption schemes as compared to purely chaos-based schemes or other number theory-based approaches. According to a recent survey on color image encryption [111], only 8.65% of the proposed schemes are based on optical transforms. In order to widen the contribution of optical transform-based schemes to image encryption, certain limitations need solutions for encouraging practical implementations.

In Section Mathematical modelling of optical transforms with FRFT and its variants, we have described the categorization of fractional transform-based image encryption schemes in accordance with the strategical amalgamation of the fractional transform domain with other evolutionary methods. There are total of eight major categories T01 to T08 (one of them T07 having two subcategories). In Figure 7, we have shown the relative contributions in terms of the number of papers published in each of these categories so far. We observe that the major contributions come from the T07: Double Image/Multiple Image category, followed by T02: chaos-based, T08: Watermarking, T03: Compressive Sensing, T01: Reality Preserving category T04: Multiple/fixed parameter transforms, T05: DNA Sequences, and least in T06: Cellular Automata.

Based on the observations related to security levels and vulnerabilities mentioned in Tables 8, 9, we elaborate on the possible ways to overcome some limitations. Most of the algorithms mainly lack in the following aspects: (1) uniform histograms, (2) entropy measure, (3) smaller key space, (4) differential analysis, (5) classical attack analysis, (6) speed analysis. In the discussion below, we try to highlight some of the possible solutions as:

- *Uniform Histogram:* A majority of fractional transform-based image encryption schemes produce cipher images having Gaussian distribution like histograms [23, 29, 141]. It is due to the fact that the energy of a transform is concentrated at the center. Authors have claimed the robustness of encryption schemes only on the basis of similarity in the distribution of histograms irrespective of the content of the plain image. The entropy measure for such distributions has values that is significantly less than the ideal value (8 for 256 intensity levels image). However, in cryptography, it is expected that the cipher image pixels should have a uniform distribution over the entire intensity ranges having entropy measure very near or equal to the ideal value. This points to some information leakage, that can make a scheme vulnerable to entropy attacks. To overcome such limitation, a hybrid algorithm in which fractional integral transform domains are amalgamated with chaos based pseudorandom substitutions should be used.
- *Smaller Key space:* Adopting multiple layer security for image encryption algorithm will lead to an increase in key space. Apart from this, making a selection of transform orders to depend on some chaotic parameters or a similar analogy will result in larger key space [141, 190]. Most of the proposed schemes have added a permutation layer along with the transform domain. Some of the schemes that are based on permutation and substitution paradigm are able to offer larger key space to overcome brute force attack.
- *Differential Analysis:* In order to fulfill the requirement of effective encryption algorithm, the scheme should be able to resist differential attack analysis. The parameters NPCR and UACI are its measures. From Table 9, it is clear that majority of schemes lack such analysis. Even if done, the UACI values are not optimum or even *zero*. This is due to the fact that there is no significant change in intensity values with a single pixel change in input. Therefore, for a successful strategy, the change should be diffused over

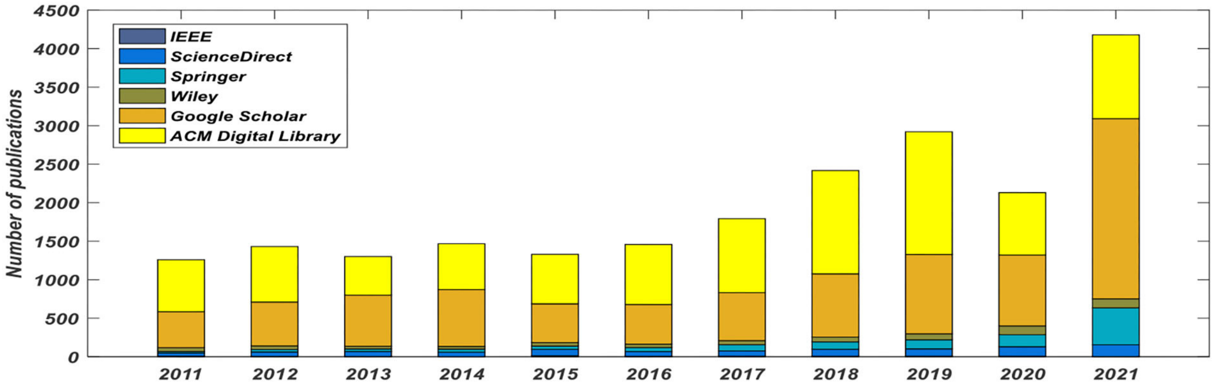


FIGURE 6
Number of papers published on fractional integral transform-based image encryption schemes on various online databases.

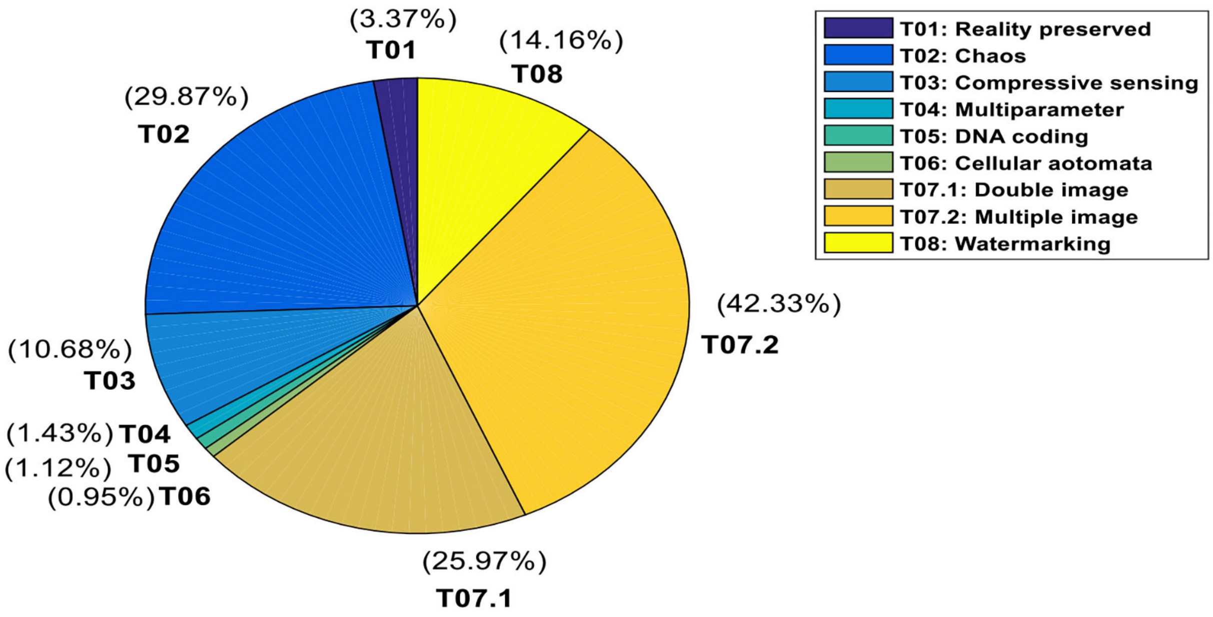


FIGURE 7
Relative contribution in terms of the number of papers published belonging to different categories (T01-T08) of fractional transform-based image encryption techniques.

the entire image coefficients. One of the solutions to this issue is to make the initial parameters of diffusion scheme to depend on some significant feature of the input image like mean or average values.

- *Time analysis:* A run time for an encryption algorithm refers to the time required for its execution. Various factors need to be considered for time analysis like the size of image, system configuration, programming language etc. [109]. To compare the computational performance of an algorithm, is a crucial task as different host machines have their own set of configurations. Due to this reason, some researchers have used an average time Vs size paradigm to evaluate computational performance [143] wherein input images with variable size are selected and the average time of encryption is evaluated using large set of different keys. Fractional transform-based encryption schemes have inherent advantage of high speed and parallel processing. However, while merging of these schemes with other domains like chaos etc., computational optimization should be taken care of. In summary, there should be trade-off management between complexity and security while designing an algorithm and some optimum suggestion for the choice of parameters, number of rounds etc. should be given.

- *Careful Selection of chaotic maps:* The chaotic maps wherever used in an encryption scheme, need a careful selection. As most of the schemes that are reviewed have employed one dimensional chaotic map [28, 66, 69]. Although 1D maps are simplest in hardware implementation but are less secure. For instance, 1D logistic maps have some periodic windows in the chaotic range [196] and that Arnold transform also has periodicity [197], hence are vulnerable. At the same time, the higher dimensional chaotic maps are sometimes secure but complex. To keep a balance, it is recommended to use a coupled map scheme where two or more 1D chaotic maps are coupled for enhanced security [50] and also robust chaotic maps may be used with proper specification of the range of parameters where robust chaos is observed. Prior to selection of such chaotic map, a proper bifurcation analysis and investigation of dynamical behavior in the entire parameter space must be done to identify the suitable regions of parameter space exhibiting robust chaos.

Conclusion

The evolution of digital media over the past two decades has revolutionized the development of strategies pertaining to security preservation of the multimedia contents. Encryption is the most effective way to secure the data. It has been observed in the study that out of all the data types, (audio, video, text, image) image data are most frequently used to convey the information. Consequently, the percentage of published work on image encryption is dominating with 42% of all the metadata available. However, cryptography for image data is challenging when it comes to classical methods of encryption due to huge volume of data and also due to the high correlation among adjacent pixel values. Various research works have been proposed in the literature that are specifically suitable for image encryption. Application of fractional integral transforms in image encryption has been an active research area and the review work in this paper is also focused on the same. The fractional integral transform provide an extra degree of freedom to the encrypted data as the fractional order of the transform is used as secret key.

The aim of this review is to build an understanding of the reader toward application of fractional integral transforms in image encryption. The initial description of the paper gives a conceptual idea on using these transforms and also the domain-based taxonomy to classify various existing schemes in the literature. The optical image encryption that comprises of optical setup and double random phase encoding (DRPE) has been discussed. Few recent review works and cryptanalysis of these schemes are tabulated and analyzed. The digital implementation of the fractional integral transforms is discussed with its analogy to the optical setup. Further, various algorithms are categorized in accordance with their merging techniques and a comprehensive review is presented on some of the most recently published articles. The performance criteria and standards to be followed have been discussed. A performance comparison in tabular format is presented for objective as well as subjective metrics of some of the recent publications. Finally, based on the observations, some major concerns are listed and a few constructive guidelines are provided. This work intends to provide the readers with an understanding of why and how fractional integral transformations are applicable to the encryption of images. In addition, the study highlights some vulnerabilities and threats associated with the usage of fractional transforms along with the probable solutions that may help in the future design and development of hybrid and robust encryption schemes.

Author contributions

GK and VP: concept and design of the article, data collection, and review. GK, RA, and VP: analysis, interpretation, and writing of the manuscript. VP: critical review and revision of the manuscript. All authors contributed to the article and approved the submitted version.

Acknowledgments

VP acknowledges the MATRICS Grant (MTR/000203/2018) from SERB, DST, Govt. of India.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Petrás I. *Fractional Derivatives, Fractional Integrals, and Fractional Differential Equations in Matlab*. London, UK: INTECH Open Access Publisher (2011).

2. Namias V. The fractional order Fourier transform and its application to quantum mechanics. *IMA J Appl Math.* (1980) 25:241–65. doi: 10.1093/imamat/25.3.241

3. McBride AC, Kerr FH. On Namias's fractional Fourier transforms. *IMA J Appl Math.* (1987) 39:159–75. doi: 10.1093/imamat/39.2.159

4. Lohmann AW. Image rotation, Wigner rotation, and the fractional Fourier transform. *JOSA A.* (1993) 10:2181–6. doi: 10.1364/JOSAA.10.002181

5. Almeida L. The fractional Fourier transform and time-frequency representations. *IEEE Transac Signal Process.* (1994) 42:3084–91. doi: 10.1109/78.330368

6. Mendelovic D, Ozaktas HM. Fractional Fourier transforms and their optical implementation. *J Opt Soc Am A.* (1993) 10:1875–81. doi: 10.1364/JOSAA.10.001875

7. Ozaktas HM, Mendlovic D. Fractional Fourier transforms and their optical implementation. *II JOSA A.* (1993) 10:2522–31. doi: 10.1364/JOSAA.10.002522

8. Ozaktas HM, Mendlovic D. Fourier transforms of fractional order and their optical interpretation. *Opt Commun.* (1993) 101:163–9. doi: 10.1016/0030-4018(93)90359-D

9. Ozaktas HM, Arikan O, Kutay MA, Bozdagt G. Digital computation of the fractional Fourier transform. *IEEE Transac Signal Process.* (1996) 44:2141–50. doi: 10.1109/78.536672

10. Candan C, Kutay MA, Ozaktas HM. The discrete fractional Fourier transform. *IEEE Transac Signal Process.* (2000) 48:1329–37. doi: 10.1109/78.839980

11. Abu Arqub O. Numerical simulation of time-fractional partial differential equations arising in fluid flows via reproducing Kernel method. *Int J Num Methods Heat Fluid Flow.* (2020) 30:4711–33. doi: 10.1108/HFF-10-2017-0394

12. Pei SC, Yeh MH, Tseng CC. Discrete fractional Fourier transform based on orthogonal projections. *IEEE Transac Signal Process.* (1999) 47:1335–48. doi: 10.1109/78.757221

13. Dickinson B, Steiglitz K. Eigenvectors and functions of the discrete Fourier transform. *IEEE Transac Acoust Speech Signal Process.* (1982) 30:25–31. doi: 10.1109/TASSP.1982.1163843

14. Pei SC, Yeh MH. Two dimensional discrete fractional Fourier transform. *Signal Process.* (1998) 67:99–108. doi: 10.1016/S0165-1684(98)00024-3

15. Pei SC, Ding JJ. Closed-form discrete fractional and affine Fourier transforms. *IEEE Transac Signal Process.* (2000) 48:1338–53. doi: 10.1109/78.839981

16. Pei SC, Yeh MH. Improved discrete fractional Fourier transform. *Opt Lett.* (1997) 22:1047–9. doi: 10.1364/OL.22.001047

17. Pei SC, Tseng CC, Yeh MH, Shyu JJ. Discrete fractional Hartley and Fourier transforms. *IEEE Transac Circ Syst II Analog Digital Signal Process.* (1998) 45:665–75. doi: 10.1109/82.686685

18. Pei SC, Yeh MH. The discrete fractional cosine and sine transforms. *IEEE Transac Signal Process.* (2001) 49:1198–207. doi: 10.1109/78.923302

19. Pei, S. C, Yeh, M. H. (1999). Discrete fractional Hadamard transform in ISCAS'99. In: *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems VLSI (Cat No 99CH36349)*. Piscataway, NJ: IEEE. 3, 179–182

20. Pei SC, Ding JJ. Fractional cosine, sine, and Hartley transforms. *IEEE Transac Signal Process.* (2002) 50:1661–80. doi: 10.1109/TSP.2002.1011207

21. Refregier P, Javidi B. Optical image encryption based on input plane and Fourier plane random encoding. *Opt Lett.* (1995) 20:767–9. doi: 10.1364/OL.20.000767

22. Unnikrishnan G, Joseph J, Singh K. Optical encryption by double-random phase encoding in the fractional Fourier domain. *Opt Lett.* (2000) 25:887–9. doi: 10.1364/OL.25.000887

23. Azoug SE, Bouguezel S. A non-linear preprocessing for opto-digital image encryption using multiple-parameter discrete fractional Fourier transform. *Opt Commun.* (2016) 359:85–94. doi: 10.1016/j.optcom.2015.09.054

24. Xiong Y, He A, Quan C. Cryptoanalysis on optical image encryption systems based on the vector decomposition technique in the Fourier domain. *Appl Opt.* (2019) 58:3301–9. doi: 10.1364/AO.58.003301

25. Maan P, Singh H. Non-linear cryptosystem for image encryption using radial Hilbert mask in fractional Fourier transform domain. *3D Res 9.* (2018) 53. doi: 10.1007/s13319-018-0205-8

26. Santhanam B, McClellan JH. The discrete rotational Fourier transform. *IEEE Transac Signal Process.* (1996) 44:994–8. doi: 10.1109/78.492554

27. Farah MB, Guesmi R, Kachouri A, Samet M. A novel chaos based optical image encryption using fractional Fourier transform and DNA sequence operation. *Opt Laser Technol.* (2020) 121:105777. doi: 10.1016/j.optlastec.2019.105777

28. Mishra DC, Sharma RK, Suman S, Prasad A. Multi-layer security of color image based on chaotic system combined with RP2DFRFT and Arnold transform. *J Inf Secur Appl.* (2017) 37:65–90. doi: 10.1016/j.jisa.2017.09.006

29. Kaur G, Agarwal R, Patidar V. Double image encryption based on 2D discrete fractional Fourier transform and piecewise nonlinear chaotic map. In: *International Conference on Advanced Informatics for Computing Research.* Singapore: Springer (2018). p. 519–30.

30. Chen B, Yu M, Tian Y, Li L, Wang D, Sun X, et al. Multiple-parameter fractional quaternion Fourier transform and its application in colour image encryption. *IET Image Process.* (2018) 12:2238–49. doi: 10.1049/iet-ipr.2018.5440

31. Lang J. Color image encryption based on color blend and chaos permutation in the reality-preserving multiple-parameter fractional Fourier transform domain. *Opt Commun.* (2015) 338:181–92. doi: 10.1016/j.optcom.2014.10.049

32. Ran Q, Yuan L, Zhao T. Image encryption based on nonseparable fractional Fourier transform and chaotic map. *Opt Commun.* (2015) 348:43–9. doi: 10.1016/j.optcom.2015.03.016

33. Zhang D, Liao X, Yang B, Zhang Y. A fast and efficient approach to color-image encryption based on compressive sensing and fractional Fourier transform. *Multimed Tools Appl.* (2018) 77:2191–208. doi: 10.1007/s11042-017-4370-1

34. Yi J, Tan G. Optical compression and encryption system combining multiple measurement matrices with fractional Fourier transform. *Appl Opt.* (2015) 54:10650–8. doi: 10.1364/AO.54.010650

35. Cariolaro G, Erseghe T, Kraniauskas P, Laurenti N. Multiplicity of fractional Fourier transforms and their relationships. *IEEE Transac Signal Process.* (2000) 48:227–41. doi: 10.1109/78.815493

36. Lang J, Tao R, Wang Y. The discrete multiple-parameter fractional Fourier transform. *Science China Inf Sci.* (2010) 53:2287–99. doi: 10.1007/s11432-010-4095-5

37. Tao R, Meng XY, Wang Y. Transform order division multiplexing. *IEEE Transac Signal Process v.* (2010) 59:598–609. doi: 10.1109/TSP.2010.2089680

38. Kang X, Zhang F, Tao R. Multichannel random discrete fractional Fourier transform. *IEEE Signal Process Lett.* (2015) 22:1340–4. doi: 10.1109/LSP.2015.2402395

39. Joshi AB, Kumar D, Gaffar A, Mishra DC. Triple color image encryption based on 2D multiple parameter fractional discrete Fourier transform and 3D Arnold transform. *Opt Lasers Eng.* (2020) 133:106139. doi: 10.1016/j.optlaseng.2020.106139

40. Pei SC, Hsue WL. The multiple-parameter discrete fractional Fourier transform. *IEEE Signal Process Lett.* (2006) 13:329–32. doi: 10.1109/LSP.2006.871721

41. Ren G, Han J, Zhu H, Fu J, Shan M. High security multiple-image encryption using discrete cosine transform and discrete multiple-parameter fractional fourier transform. *J Commun.* (2016) 11:491–7. doi: 10.12720/jcm.11.5.491-497

42. Tomassini M, Sipper M, Perrenoud M. On the generation of high-quality random numbers by two-dimensional cellular automata. IEEE *Transac computers.* (2000) 49:1146–51. doi: 10.1109/12.888056

43. Sui L, Duan K, Liang J, Zhang Z, Meng H. Asymmetric multiple-image encryption based on coupled logistic maps in fractional Fourier transform domain. *Opt Laser Eng.* (2014) 62:139–52. doi: 10.1016/j.optlaseng.2014.06.003

44. Ran Q, Zhang H, Zhang J, Tan L, Ma J. Deficiencies of the cryptography based on multiple-parameter fractional Fourier transform. *Opt Lett.* (2009) 34:1729–31. doi: 10.1364/OL.34.001729

45. Zhao T, Ran Q, Yuan L, Chi Y, Ma J. Security of image encryption scheme based on multi-parameter fractional Fourier transform. *Opt Commun.* (2016) 376:47–51. doi: 10.1016/j.optcom.2016.05.016

46. Youssef A. On the security of a cryptosystem based on multiple-parameters discrete fractional Fourier transform. *IEEE Signal Process Letters.* (2008) 15:77–8. doi: 10.1109/LSP.2007.910299

47. Wu J, Guo F, Liang Y, Zhou N. Triple color images encryption algorithm based on scrambling and the reality-preserving fractional discrete cosine transform. *Optik.* (2014) 125:4474–9. doi: 10.1016/j.ijleo.2014.02.026

48. Wu J, Zhang M, Zhou N. Image encryption scheme based on random fractional discrete cosine transform and dependent scrambling and diffusion. *J Modern Opt.* (2017) 64:334–46. doi: 10.1080/09500340.2016.1236990

49. Pan SM, Wen RH, Zhou ZH, Zhou NR. Optical multi-image encryption scheme based on discrete cosine transform and nonlinear fractional Mellin transform. *Multimed Tools Appl.* (2017) 76:2933–53. doi: 10.1007/s11042-015-3209-x

50. Faragallah OS. Optical double color image encryption scheme in the Fresnel-based Hartley domain using Arnold transform and chaotic logistic adjusted sine phase masks. *Opt Quant Electron.* (2018) 50:118. doi: 10.1007/s11082-018-1363-x

51. Yadav AK, Singh P, Singh K. Cryptosystem based on devil's vortex Fresnel lens in the fractional Hartley domain. *J Opt.* (2018) 47:208–19. doi: 10.1007/s12596-017-0435-9

52. Kaur G, Agarwal R, Patidar V. Multiple image encryption with fractional hartley transform and robust chaotic mapping. In: *2019 6th International Conference on Signal Processing and Integrated Networks (SPIN).* (2019). Piscataway, NJ: IEEE. 399–403. doi: 10.1109/SPIN.2019.8711777

53. Kaur G, Agarwal R, Patidar V. Color image encryption system using combination of robust chaos and chaotic order fractional Hartley transformation. *J King Saud Univ Comput Inf Sci.* (2021) 34:5883–97. doi: 10.1016/j.jksuci.2021.03.007

54. Kaur G, Agarwal R, Patidar V. Color image encryption scheme based on fractional Hartley transform and chaotic substitution–permutation. *Visual Comput.* (2021) 38:1027–50. doi: 10.1007/s00371-021-02066-w

55. Wang M, Pousset Y, Carré P, Perrine C, Zhou N, Wu J, et al. Optical image encryption scheme based on apertured fractional Mellin transform. *Opt Laser Technol.* (2020) 124:106001. doi: 10.1016/j.optlastec.2019.106001

56. Zhou N, Wang Y, Gong L. Novel optical image encryption scheme based on fractional Mellin transform. *Opt Commun.* (2011) 284:3234–42. doi: 10.1016/j.optcom.2011.02.065

57. Zhou N, Wang Y, Gong L, Chen X, Yang Y. Novel color image encryption algorithm based on the reality preserving fractional Mellin transform. *Opt Laser Technol.* (2012) 44:2270–81. doi: 10.1016/j.optlastec.2012.02.027

58. Zhou N, Li H, Wang D, Pan S, Zhou Z. Image compression and encryption scheme based on 2D compressive sensing and fractional Mellin transform. *Opt Commun.* (2015) 343:10–21. doi: 10.1016/j.optcom.2014.12.084

59. Wang M, Pousset Y, Carré P, Perrine C, Zhou N, Wu J, et al. Image encryption scheme based on a Gaussian apertured reality-preserving fractional Mellin transform. *Optica Applicata.* (2020) 50:477–495. /*oa*200312 doi: 10.37190/oa200312

60. Kang X, Ming A, Tao R. Reality-preserving multiple parameter discrete fractional angular transform and its application to color image encryption. *IEEE Transac Circ Syst Video Technol.* (2018) 29:1595–607. doi: 10.1109/TCSVT.2018.2851983

61. Tong LJ, Zhou NR, Huang ZJ, Xie XW, Liang YR. Nonlinear multi-image encryption scheme with the reality-preserving discrete fractional angular transform and DNA sequences. *Secur Commun Netw.* (2021) 2021:1–18. doi: 10.1155/2021/6650515

62. Yu J, Li Y, Xie X, Zhou N, Zhou Z. Image encryption algorithm by using the logistic map and discrete fractional angular transform. *Optica Applicata.* (2017) 47. doi: 10.5277/oa17011310.5277/oa170113

63. Sui L, Duan K, Liang J. Double-image encryption based on discrete multiple-parameter fractional angular transform and two-coupled logistic maps. *Opt Commun.* (2016) 343:140–9. doi: 10.1016/j.optcom.2015.01.021

64. Sui L, Duan K, Liang J. A secure double-image sharing scheme based on Shamir's three-pass protocol and 2D Sine Logistic modulation map in discrete multiple-parameter fractional angular transform domain. *Opt Laser Eng.* (2016) 80:52–62. doi: 10.1016/j.optlaseng.2015.12.016

65. Vilardy JM, Millán MS, Pérez-Cabré E. Nonlinear image encryption using a fully phase nonzero-order joint transform correlator in the Gyrator domain. *Opt Laser Eng.* (2017) 89:88–94. doi: 10.1016/j.optlaseng.2016.02.013

66. Abuturab M. Securing color information using Arnold transform in gyrator transform domain. *Opt Laser Eng.* (2012) 50:772–9. doi: 10.1016/j.optlaseng.2011.12.006

67. Singh H, Yadav AK, Vashisth S, Singh K. Double phase-image encryption using gyrator transforms, and structured phase mask in the frequency plane. *Opt Laser Eng.* (2015) 67:145–56. doi: 10.1016/j.optlaseng.2014.10.011

68. Singh N, Sinha A. Gyrator transform-based optical image encryption, using chaos. *Opt Laser Eng.* (2009) 47:539–46. doi: 10.1016/j.optlaseng.2008.10.013

69. Abuturab M. Group multiple-image encoding and watermarking using coupled logistic maps and gyrator wavelet transform. *JOSA A.* (2015) 32:1811–20. doi: 10.1364/JOSAA.32.001811

70. Li H, Wang Y, Yan H, Li L, Li Q, Zhao X, et al. Double-image encryption by using chaos-based local pixel scrambling technique and gyrator transform. *Opt Laser Eng.* (2013) 51:1327–31. doi: 10.1016/j.optlaseng.2013.05.011

71. Shao Z, Duan Y, Coatrieux G, Wu J, Meng J, Shu H, et al. Combining double random phase encoding for color image watermarking in quaternion gyrator domain. *Opt Commun.* (2015) 343:56–65. doi: 10.1016/j.optcom.2015.01.002

72. Belazi A, El-Latif AAA, Diaconu AV, Rhouma R, Belghith S. Chaos-based partial image encryption scheme based on linear fractional and lifting wavelet transforms. *Opt Laser Eng.* (2017) 88:37–50. doi: 10.1016/j.optlaseng.2016.07.010

73. Javidi, B. E. (2006). *Optical and Digital Techniques for Information Security (Vol 1).* Berlin, Germany: Springer Science and Business Media.

74. Unnikrishnan G, Joseph J. and K.Singh. *Optical encryption system that uses phase conjugation in a photorefractive crystal Appl Opt.* (1998) 37:8181–6. doi: 10.1364/AO.37.008181

75. Weimann S, Perez-Leija A, Lebugle M, Keil R, Tichy M, Gräfe M, et al. (2016). Implementation of quantum and classical discrete fractional Fourier transforms. *Nat Commun.* 7, 1–8. doi: 10.1038/ncomms11027

76. Javidi B, Nomura T. Securing information by use of digital holography. *Opt Lett.* (2000) 25:28–30. doi: 10.1364/OL.25.000028

77. Tajahuerce E, Javidi B. Encrypting three-dimensional information with digital holography. *Appl Opt.* (2000) 39:6595–601. doi: 10.1364/AO.39.006595

78. Kim M. Principles and techniques of digital holographic microscopy. *SPIE Rev.* (2010) 1:018005. doi: 10.1117/6.0000006

79. Osten, W, Faridian, A, Gao, P, Körner, K, Naik, D, Pedrini, G. (2014). Recent advances in digital holography. *Appl Opt.* 53. G44–G63. doi: 10.1364/AO.53.000G44

80. Shi Y, Li T, Wang Y, Gao Q, Zhang S, Li H, et al. Optical image encryption via ptychography. *Opt Lett.* (2013) 38:1425–7. doi: 10.1364/OL.38.001425

81. Rawat N, Hwang IC, Shi Y, Lee BG. Optical image encryption via photon-counting imaging and compressive sensing based ptychography. *J Opt.* (2015) 17:065704. doi: 10.1088/2040-8978/17/6/065704

82. Gao Q, Wang Y, Li T, Shi Y. Optical encryption of unlimited-size images based on ptychographic scanning digital holography. *Appl Opt.* (2014) 53:4700–7. doi: 10.1364/AO.53.004700

83. Su Y, Xu W, Zhao J. Optical image encryption based on chaotic fingerprint phase mask and pattern-illuminated Fourier ptychography. *Opt Laser Eng.* (2020) 128:106042. doi: 10.1016/j.optlaseng.2020.106042

84. Liu L, Shan M, Zhong Z, Liu B. Multiple-image encryption and authentication based on optical interference by sparsification and space multiplexing. *Opt Laser Technol.* (2020) 122:105858. doi: 10.1016/j.optlastec.2019.105858

85. Clemente P, Durán V, Tajahuerce E, Lancis J. Optical encryption based on computational ghost imaging. *Opt Lett.* (2010) 35:2391–3. doi: 10.1364/OL.35.002391

86. Yi K, Leihong Z, Hualong Y, Mantong Z, Kanwal S, Dawei Z, et al. Camouflaged optical encryption based on compressive ghost imaging. *Opt Laser Eng.* (2020) 134:106154. doi: 10.1016/j.optlaseng.2020.106154

87. Du J, Xiong Y, Quan C. High-efficiency optical image authentication scheme based on ghost imaging and block processing. *Opt Commun.* (2020) 460:125113. doi: 10.1016/j.optcom.2019.125113

88. Chen W, Chen X, Sheppard CJ. Optical image encryption based on diffractive imaging. *Opt Lett.* (2010) 35:3817–9. doi: 10.1364/OL.35.003817

89. Qin Y, Wang Z, Pan Q, Gong Q. Optical color-image encryption in the diffractive-imaging scheme. *Opt Laser Eng.* (2016) 77:191–202. doi: 10.1016/j.optlaseng.2015.09.002

90. He X, Tao H, Zhang L, Yuan X, Liu C, Zhu J, et al. Single-Shot optical multiple-image encryption based on polarization-resolved diffractive imaging. *IEEE Photon J.* (2019) 11:1–12. doi: 10.1109/JPHOT.2019.2939164

91. Hazer A, Yildirim R. Hiding data with simplified diffractive imaging based hybrid method. *Opt Laser Technol.* (2020) 128:106237. doi: 10.1016/j.optlastec.2020.106237

92. Gopinathan U, Naughton TJ, Sheridan JT. Polarization encoding and multiplexing of two-dimensional signals: application to image encryption. *Appl Opt.* (2006) 45:5693–700. doi: 10.1364/AO.45.005693

93. Wang Q, Xiong D, Alfalou A, Brosseau C. Optical image encryption method based on incoherent imaging and polarized light encoding. *Opt Commun.* (2018) 415:56–63. doi: 10.1016/j.optcom.2018.01.018

94. Nomura T, Javidi B. Optical encryption using a joint transform correlator architecture. *Opt Eng.* (2000) 39:2031–5. doi: 10.1117/1.1304844

95. Zhao H, Zhong Z, Fang W, Xie H, Zhang Y, Shan M. Double-image encryption using chaotic maps and nonlinear non-DC joint fractional Fourier transform correlator. *Opt Eng.* (2016) 55:093109. doi: 10.1117/1.OE.55.9.093109

96. Chen W. Optical cryptosystem based on single-pixel encoding using the modified Gerchberg–Saxton algorithm with a cascaded structure. *JOSA A.* (2016) 33:2305–11. doi: 10.1364/JOSAA.33.002305

97. Guo C, Wei C, Tan J, Chen K, Liu S, Wu Q, et al. A review of iterative phase retrieval for measurement and encryption. *Opt Laser Eng.* (2017) 89:2–12. doi: 10.1016/j.optlaseng.2016.03.021

98. Huang H, Yang S, Ye R. Image encryption scheme combining a modified Gerchberg–Saxton algorithm with hyper-chaotic system. *Soft Comput.* (2019) 23:7045–53. doi: 10.1007/s00500-018-3345-0

99. Hennelly B, Sheridan JT. Optical image encryption by random shifting in fractional Fourier domains. *Opt Lett.* (2003) 28:269–71. doi: 10.1364/OL.28.000269

100. Liansheng S, Cong D, Minjie X, Ailing T, Anand A. Information encryption based on the customized data container under the framework of computational ghost imaging. *Opt Expr.* (2019) 27:16493–506. doi: 10.1364/OE.27.016493

101. Moreno I, Ferreira C. 3 Fractional Fourier Transforms and Geometrical Optics. *Adv Imag Electr Phys.* (2010) 161:89. doi: 10.1016/S1076-5670(10)61003-8

102. Chen W, Javidi B, Chen X. Advances in optical security systems. *Adv Opt Photon.* (2014) 6:120–55. doi: 10.1364/AOP.6.000120

103. Javidi B, Carnicer A, Yamaguchi M, Nomura T, Pérez-Cabré E Millán MS, et al. Roadmap on optical security. *J Opt.* (2016) 18:083001. doi: 10.1088/2040-8978/18/8/083001

104. Sejdić E, Djurović I, Stanković,L (2011). Fractional Fourier transform as a signal *process.* tool: an overview of recent developments. *Signal Process.* (2011) 91:1351–69. doi: 10.1016/j.sigpro.2010.10.008

105. Saxena R, Singh K. Fractional Fourier transform: a novel tool for signal processing. *J Indian Inst Sci.* (2013) 85:11.

106. Yang Q, Chen D, Zhao T, Chen Y. Fractional calculus in image processing: a review. *Frac Calc Appl Anal.* (2016) 19:1222–49. doi: 10.1515/fca-2016-0063

107. Guo C, Muniraj I, Sheridan JT. Phase-retrieval-based attacks on linear-canonical-transform-based DRPE systems. *Appl Opt.* (2016) 55:4720–8. doi: 10.1364/AO.55.004720

108. Situ GH, Wang HC. Phase problems in optical imaging. *Front Inf Technol Electron Eng.* (2017) 18:1277–87. doi: 10.1631/FITEE.1700298

109. Kaur M, Kumar V. A comprehensive review on image encryption techniques. *Arch Comput Methods Eng.* (2020) 27:15–43. doi: 10.1007/s11831-018-9298-8

110. Jinming M, Hongxia M, Xinhua S, Chang G, Xuejing K, Ran T, et al. Research progress in theories and applications of the fractional Fourier transform. *Opto-Electron Eng.* (2018) 45:170747.

111. Ghadirli HM, Nodehi A, Enayatifar R. An overview of encryption algorithms in color images. *Signal Process.* (2019) 164:163–85. doi: 10.1016/j.sigpro.2019.06.010

112. Jindal N, Singh K. Applicability of fractional transforms in image processing-review, technical challenges and future trends. *Multimedia Tools Appl.* (2019) 78:10673–700. doi: 10.1007/s11042-018-6594-0

113. Gómez-Echavarría A, Ugarte JP, Tobón C. The fractional Fourier transform as a biomedical signal and image processing tool: a review. *Biocybern Biomed Eng.* (2020) 40:1081–93. doi: 10.1016/j.bbe.2020.05.004

114. Abd-El-Atty B, Iliyasu AM, Alanezi A, Abd El-latif AA. Optical image encryption based on quantum walks. *Opt Lasers Eng.* (2021) 138:106403. doi: 10.1016/j.optlaseng.2020.106403

115. Zhou K, Fan J, Fan H, Li M. Secure image encryption scheme using double random-phase encoding and compressed sensing. *Opt Laser Technol.* (2020) 121:105769. doi: 10.1016/j.optlastec.2019.105769

116. Huang ZJ, Cheng S, Gong LH, Zhou NR. Nonlinear optical multi-image encryption scheme with two-dimensional linear canonical transform. *Opt Laser Eng.* (2020) 124:105821. doi: 10.1016/j.optlaseng.2019.105821

117. Huo D, Zhou DF, Yuan S, Yi S, Zhang L, Zhou X, et al. Image encryption using exclusive-OR with DNA complementary rules and double random phase encoding. *Phys Lett A.* (2019) 383:915–22. doi: 10.1016/j.physleta.2018.12.011

118. Gong L, Qiu K, Deng C, Zhou N. An optical image compression and encryption scheme based on compressive sensing and RSA algorithm. *Opt Laser Eng.* (2019) 121:169–80. doi: 10.1016/j.optlaseng.2019.03.006

119. Chen H, Liu Z, Zhu L, Tanougast C, Blondel W. Asymmetric color cryptosystem using chaotic Ushiki map and equal modulus decomposition in fractional Fourier transform domains. *Opt Laser Eng.* (2019) 112:7–15. doi: 10.1016/j.optlaseng.2018.08.020

120. Kumar R, Sheridan JT, Bhaduri B. Nonlinear double image encryption using 2D non-separable linear canonical transform and phase retrieval algorithm. *Opt Laser Technol.* (2018) 107:353–60. doi: 10.1016/j.optlastec.2018.06.014

121. Jiao S, Zou W, Li X. QR code based noise-free optical encryption and decryption of a gray scale image. *Opt Commun.* (2017) 387:235–40. doi: 10.1016/j.optcom.2016.11.066

122. Khurana M, Singh H. An asymmetric image encryption based on phase truncated hybrid transform. *3D Res 8.* (2017) 28. doi: 10.1007/s13319-017-0137-8

123. Su Y, Tang C, Chen X, Li B, Xu W, Lei Z, et al. Cascaded Fresnel holographic image encryption scheme based on a constrained optimization algorithm and Henon map. *Opt Laser Eng.* (2017) 88:20–7. doi: 10.1016/j.optlaseng.2016.07.012

124. Li X, Li C, Lee IK. Chaotic image encryption using pseudo-random masks and pixel mapping. *Signal Process.* (2016) 125:48–63. doi: 10.1016/j.sigpro.2015.11.017

125. Yuan S, Yao J, Liu X, Zhou X, Li Z. Cryptanalysis and security enhancement of optical cryptography based on computational ghost imaging. *Opt Commun.* (2016) 365:180–5. doi: 10.1016/j.optcom.2015.12.013

126. Li T, Shi Y. Vulnerability of impulse attack-free four random phase mask cryptosystems to chosen-plaintext attack. *J Opt.* (2016) 18:035702. doi: 10.1088/2040-8978/18/3/035702

127. Wang Y, Quan C, Tay CJ. Cryptanalysis of an information encryption in phase space. *Opt Laser Eng.* (2016) 85:65–71. doi: 10.1016/j.optlaseng.2016.04.024

128. Liao M, He W, Lu D, Peng X. Ciphertext-only attack on optical cryptosystem with spatially incoherent illumination: from the view of imaging through scattering medium. *Sci Rep.* (2017) 7:41789. doi: 10.1038/srep41789

129. Hai H, Pan S, Liao M, Lu D, He W, Peng X, et al. Cryptanalysis of random-phase-encoding-based optical cryptosystem via deep learning. *Opt Expr.* (2019) 27:21204–13. doi: 10.1364/OE.27.021204

130. Xiong Y, He A, Quan C. Cryptanalysis of an optical cryptosystem based on phase-truncated Fourier transform and nonlinear operations. *Opt Commun.* (2018) 428:120–30. doi: 10.1016/j.optcom.2018.07.058

131. Dou S, Shen X, Zhou B, Lin C, Huang F, Lin Y, et al. Known-plaintext attack on JTC-based linear cryptosystem. *Optik.* (2019) 198:163274. doi: 10.1016/j.ijleo.2019.163274

132. Chang X, Yan A, Zhang H. Ciphertext-only attack on optical scanning cryptography. *Opt Laser Eng.* (2020) 126:105901. doi: 10.1016/j.optlaseng.2019.105901

133. Jiao S, Gao Y, Lei T, Yuan X. Known-plaintext attack to optical encryption systems with space and polarization encoding. *Opt Expr.* (2020) 28:8085–97. doi: 10.1364/OE.387505

134. Zhou L, Xiao Y, Chen W. Vulnerability to machine learning attacks of optical encryption based on diffractive imaging. *Opt Laser Eng.* (2020) 125:105858. doi: 10.1016/j.optlaseng.2019.105858

135. He W, Pan S, Liao M, Lu D, Xing Q, Peng X, et al. Cryptanalysis of phase-truncated Fourier-transforms-based optical cryptosystem using an untrained neural network. In Advanced Optical Imaging Technologies III, 115491W. *Int Soc Opt Photon.* (2020) 11549. doi: 10.1117/12.2583396 [Epub ahead of print].

136. Song W, Liao X, Weng D, Zheng Y, Liu Y, Wang Y, et al. Cryptanalysis of phase information based on a double random-phase encryption method. *Opt Commun.* (2021) 497:127172. doi: 10.1016/j.optcom.2021.127172

137. Arikan O, Kutay MA, Ozaktas HM, Akdemir OK. The discrete fractional fourier transformation. In: *Proceedings of Third International Symposium on Time-Frequency and Time-Scale Analysis (TFTS-96).* Paris: IEEE (1996). doi: 10.1109/TFSA.1996.547486

138. Belazi A, El-Latif AAA, Belghith. S. A novel image encryption scheme based on substitution-permutation network and chaos. *Signal Process.* (2016) 128:155–170. doi: 10.1016/j.sigpro.2016.03.021

139. Venturini I, Duhamel P. Reality preserving fractional transforms [signal processing applications]. In: *Acoustics, Speech, and Signal Processing, France* (2004).

140. Liang Y, Liu G, Zhou N, Wu J. Color image encryption combining a reality-preserving fractional DCT with chaotic mapping in HSI space. *Multimedia Tools Appl.* (2016) 75:6605–20. doi: 10.1007/s11042-015-2592-7

141. Kang X, Tao R. Color image encryption using pixel scrambling operator and reality-preserving MPFRHT. *IEEE Transac Circ Syst Video Technol.* (2018) 29:1919–32. doi: 10.1109/TCSVT.2018.2859253

142. Alvarez G, Li S. Some basic cryptographic requirements for chaos-based cryptosystems. *Int J Bifur Chaos.* (2006) 16:2129–51. doi: 10.1142/S0218127406015970

143. Patidar V, Pareek NK, Purohit G, Sud KK. A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption. *Opt Commun.* (2011) 284:4331–9. doi: 10.1016/j.optcom.2011.05.028

144. Rahman SMM, Hossain MA, Mouftah H, El Saddik A, Okamoto E. Chaos-cryptography based privacy preservation technique for video surveillance. *Multimedia systems.* (2012) 18:145–55. doi: 10.1007/s00530-011-0246-9

145. Fu C, Lin BB, Miao YS, Liu X, Chen JJ. A novel chaos-based bit-level permutation scheme for digital image encryption. *Opt Commun.* (2011) 284:5415–23. doi: 10.1016/j.optcom.2011.08.013

146. Zhang X, Zhao Z, Wang J. Chaotic image encryption based on circular substitution box and key stream buffer. *Signal Process Image Commun.* (2014) 29:902–13. doi: 10.1016/j.image.2014.06.012

147. Sam IS, Devaraj P, Bhuvaneswaran RS. A novel image cipher based on mixed transformed logistic maps. *Multimed Tools Appl.* (2012) 56:315–30. doi: 10.1007/s11042-010-0652-6

148. Parvaz R, Zarebnia M. A combination chaotic system and application in color image encryption. *Opt Laser Technol.* (2018) 101:30–41. doi: 10.1016/j.optlastec.2017.10.024

149. Zhu C. A novel image encryption scheme based on improved hyperchaotic sequences. *Opt Commun.* (2012) 285:29–37. doi: 10.1016/j.optcom.2011.08.079

150. Boriga R, Dăscălescu AC, Priescu I. A new hyperchaotic map and its application in an image encryption scheme. *Signal Process Image Commun.* (2014) 29:887–901. doi: 10.1016/j.image.2014.04.001

151. Li Y, Wang C, Chen H. A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. *Opt Laser Eng.* (2017) 90:238–46. doi: 10.1016/j.optlaseng.2016.10.020

152. Zhang Y, Xiao D. Double optical image encryption using discrete Chirikov standard map and chaos-based fractional random transform. *Opt Laser Eng.* (2013) 51:472–80. doi: 10.1016/j.optlaseng.2012.11.001

153. Kang X, Tao R, Zhang F. Multiple-parameter discrete fractional transform and its applications. *IEEE Transac Signal Process.* (2016) 64:3402–17. doi: 10.1109/TSP.2016.2544740

154. Chen J, Zhu ZL, Zhang LB, Zhang Y, Yang BQ. Exploiting self-adaptive permutation–diffusion and DNA random encoding for secure and efficient image encryption, Signal *Process. 142.* (2018) 340–53. doi: 10.1016/j.sigpro.2017.07.034

155. Wu J, Cao X, Liu X, Ma L, Xiong J. Image encryption using the random FrDCT and the chaos-based game of life, J. *Modern Opt.* (2019) 66:764–75. doi: 10.1080/09500340.2019.1571249

156. Jamal SS, Shah T, AlKhaldi AH, Tufail MN. Construction of new substitution boxes using linear fractional transformation and enhanced chaos. *Chin J Phys.* (2019) 60:564–72. doi: 10.1016/j.cjph.2019.05.038

157. Donoho D. Compressed sensing. *IEEE Transac Inf Theory.* (2006) 52:1289–306. doi: 10.1109/TIT.2006.871582

158. Candès E. Compressive sampling. In: *Proceedings of the International Congress of Mathematicians.* (2006). p. 1433–52. doi: 10.4171/022-3/69

159. Gong L, Qiu K, Deng C, Zhou N. An image compression and encryption algorithm based on chaotic system and compressive sensing. *Opt Laser Technol.* (2019) 115:257–67. doi: 10.1016/j.optlastec.2019.01.039

160. Lang J, Zhang J. Optical image cryptosystem using chaotic phase-amplitude masks encoding and least-data-driven decryption by compressive sensing. *Opt Commun.* (2015) 338:45–53. doi: 10.1016/j.optcom.2014.10.018

161. Zhou N, Yang J, Tan C, Pan S, Zhou Z. Double-image encryption scheme combining DWT-based compressive sensing with discrete fractional random transform. *Opt Commun.* (2015) 354:112–21. doi: 10.1016/j.optcom.2015.05.043

162. Deng J, Zhao S, Wang Y, Wang L, Wang H, Sha H, et al. Image compression-encryption scheme combining 2D compressive sensing with discrete fractional random transform. *Multimed Tools Appl.* (2017) 76:10097–117. doi: 10.1007/s11042-016-3600-2

163. Chen XD, Wang Y, Wang J, Wang QH. Asymmetric color cryptosystem based on compressed sensing and equal modulus decomposition in discrete fractional random transform domain. *Opt Laser Eng.* (2019) 121:143–9. doi: 10.1016/j.optlaseng.2019.04.004

164. Ye HS, Dai JY, Wen SX, Gong LH, Zhang WQ. Color image encryption scheme based on quaternion discrete multi-fractional random transform and compressive sensing. *Optica Applicata.* (2021) 51. doi: 10.37190/oa210304

165. Tao R, Meng XY, Wang Y. Image encryption with multiorders of fractional Fourier transforms. *IEEE Transac Inf For Secur.* (2010) 5:734–8. doi: 10.1109/TIFS.2010.2068289

166. Zhong Z, Qin H, Liu L, Zhang Y, Shan M. Silhouette-free image encryption using interference in the multiple-parameter fractional Fourier transform domain. *Opt Expr.* (2017) 25:6974–82. doi: 10.1364/OE.25.006974

167. Watson JD, Crick FH. A structure for deoxyribose nucleic acid. *Nature.* (1953) 171:737–8. doi: 10.1038/171737a0

168. Mills Jr AP, Yurke B, Platzman PM. Article for analog vector algebra computation. *Biosystems.* (1999) 52:175–80. doi: 10.1016/S0303-2647(99)00044-1

169. Wasiewicz P, Mulawka JJ, Rudnicki WR, Lesyng B. Adding numbers with DNA. In: Smc 2000 conference proceedings. IEEE international conference on systems, man and cybernetics.'cybernetics evolving to systems, humans, organization and their complex interactions'(cat. no. 0) (1, 265–270.). Piscataway, NJ: IEEE.

170. Wei R, Li X, Wang QH. Double color image encryption scheme based on off-axis holography and maximum length cellular automata. *Optik.* (2017) 145:407–17. doi: 10.1016/j.ijleo.2017.07.046

171. Li XW, Cho SJ, Kim ST. A 3D image encryption technique using computer-generated integral imaging and cellular automata transform. *Optik.* (2014) 125:2983–90. doi: 10.1016/j.ijleo.2013.12.036

172. Sui L, Lu H, Wang Z, Sun Q. Double-image encryption using discrete fractional random transform and logistic maps. *Opt Laser Eng.* (2014) 56:1–12. doi: 10.1016/j.optlaseng.2013.12.001

173. Yuan L, Ran Q, Zhao T. Image authentication based on double-image encryption and partial phase decryption in nonseparable fractional Fourier domain. *Opt Laser Technol.* (2017) 88:111–20. doi: 10.1016/j.optlastec.2016.09.004

174. Liansheng S, Cong D, Xiao Z, Ailing T, Anand A. Double-image encryption based on interference and logistic map under the framework of double random phase encoding. *Opt Laser Eng.* (2019) 122:113–1228. doi: 10.1016/j.optlaseng.2019.06.005

175. Liu W, Xie Z, Liu Z, Zhang Y, Liu S. Multiple-image encryption based on optical asymmetric key cryptosystem. *Opt Commun.* (2015) 335:205–11. doi: 10.1016/j.optcom.2014.09.046

176. Guleria V, Sabir S, Mishra DC. Security of multiple RGB images by RSA cryptosystem combined with FrDCT and Arnold transform, J. *Inf Secur Appl.* (2020) 54:102524. doi: 10.1016/j.jisa.2020.102524

177. Guo Y, Li BZ. Blind image watermarking method based on linear canonical wavelet transform and QR decomposition. *IET Image Process.* (2016) 10:773–86. doi: 10.1049/iet-ipr.2015.0818

178. Kaur G, Agarwal R, Patidar V. Crypto-watermarking of images for secure transmission over cloud, J. *Inf Optim Sci.* (2020) 41:205–16. doi: 10.1080/02522667.2020.1714185

179. Xiao B, Luo J, Bi X, Li W, Chen B. Fractional discrete Tchebyshev moments and their applications in image encryption and watermarking. *Inf Sci.* (2020) 516:545–59. doi: 10.1016/j.ins.2019.12.044

180. Singh H. Watermarking image encryption using deterministic phase mask and singular value decomposition in fractional Mellin transform domain IET Image *Process.* (2018) 12:1994–2001. doi: 10.1049/iet-ipr.2018.5399

181. Chen B, Zhou C, Jeon B, Zheng Y, Wang J. Quaternion discrete fractional random transform for color image adaptive watermarking. *Multimed Tools Appl.* (2018) 77:20809–37. doi: 10.1007/s11042-017-5511-2

182. Liu X, Han G, Wu J, Shao Z, Coatrieux G, Shu H, et al. Fractional Krawtchouk transform with an application to image watermarking. *IEEE Transac Signal Process.* (2017) 65:1894–908. doi: 10.1109/TSP.2017.2652383

183. Lian S. *Multimedia Content Encryption: Techniques and Applications.* (2008). Boca Raton, FL: CRC Press.

184. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. *IEEE Transac image process.* (2004) 13:600–12. doi: 10.1109/TIP.2003.819861

185. Zhang YQ, Wang XY. A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice. *Inf Sci.* (2014) 273:329–51. doi: 10.1016/j.ins.2014.02.156

186. Shannon C. Communication theory of secrecy systems. *Bell Syst Tech J.* (1949) 228:656–715. doi: 10.1002/j.1538-7305.1949.tb00928.x

187. Kwok HS, Tang WK. A fast image encryption system based on chaotic maps with finite precision representation. *Chaos Solitons Fractals.* (2007) 32:1518–29. doi: 10.1016/j.chaos.2005.11.090

188. Wu Y, Zhou Y, Saveriades G, Agaian S, Noonan JP, Natarajan P, et al. Local Shannon entropy measure with statistical tests for image randomness. *Inf Sci.* (2013) 222:323–42. doi: 10.1016/j.ins.2012.07.049

189. Wu Y, Noonan JP, Agaian S. NPCR and UACI randomness tests for image encryption. *Cyber J Multidiscip J Sci Technol J Select Areas Telecommun (JSAT).* (2011) 1:31–8.

190. Kaur G, Agarwal R, Patidar V. Chaos based multiple order optical transform for 2D image encryption. *Eng Sci Technol Int J.* (2020) 23:998–1014. doi: 10.1016/j.jestch.2020.02.007

191. Askar SS, Karawia AA, Al-Khedhairi A, Al-Ammar FS. An algorithm of image encryption using logistic and two-dimensional chaotic economic maps. *Entropy.* (2019) 21:44. doi: 10.3390/e21010044

192. Wu Y, Zhou Y, Noonan JP, Agaian S. Design of image cipher using latin squares, Information Sciences. *v.* (2014) 264:317–39. doi: 10.1016/j.ins.2013.11.027

193. Zhang Y, Xiang Y, Zhang LY, Rong Y, Guo S. Secure wireless communications based on compressive sensing: a survey. *IEEE Commun Surv Tutorials.* (2018) 21:1093–111. doi: 10.1109/COMST.2018.2878943

194. Ponnaian D, Chandranbabu K. Crypt analysis of an image compression–encryption algorithm and a modified scheme using compressive sensing. *Optik.* (2017) 147:263–76. doi: 10.1016/j.ijleo.2017.07.0635

195. Yang Z, Yan W, Xiang Y. On the security of compressed sensing-based signal cryptosystem. *IEEE Transac Emerg Top Comput.* (2014) 3:363–71. doi: 10.1109/TETC.2014.2372151

196. Zhou Y, Hua Z, Pun CM, Chen CP. Cascade chaotic system with applications. *IEEE Transac Cybern.* (2014) 45:2001–12. doi: 10.1109/TCYB.2014.2363168

197. Dyson FJ, Falk H. Period of a discrete cat mapping. *Am Math Monthly.* (1992) 99:603–14. doi: 10.1080/00029890.1992.11995900

frontiers | Frontiers in Applied Mathematics and Statistics

TYPE Original Research
PUBLISHED 28 October 2022
DOI 10.3389/fams.2022.1021551

Continuous Hyper-parameter OPtimization (CHOP) in an ensemble Kalman filter

Xiaodong Luo[1]* and Chuan-An Xia[2]*

[1]Technology Department, Norwegian Research Centre (NORCE), Bergen, Norway, [2]Department of Hydrology and Water Resources Environment Engineering, East China University of Technology, Nanchang, China

Check for updates

OPEN ACCESS

EDITED BY
Sergei Pereverzyev,
Johann Radon Institute for Computational and Applied Mathematics (RICAM), Austria

REVIEWED BY
Bruno Veloso,
Portucalense University, Portugal
Xin Tong,
National University of Singapore, Singapore

*CORRESPONDENCE
Xiaodong Luo
xluo@norceresearch.no
Chuan-An Xia
c.a.xia@cugb.edu.cn

SPECIALTY SECTION
This article was submitted to Mathematics of Computation and Data Science,
a section of the journal
Frontiers in Applied Mathematics and Statistics

RECEIVED 17 August 2022
ACCEPTED 28 September 2022
PUBLISHED 28 October 2022

CITATION
Luo X and Xia C-A (2022) Continuous Hyper-parameter OPtimization (CHOP) in an ensemble Kalman filter. *Front. Appl. Math. Stat.* 8:1021551.
doi: 10.3389/fams.2022.1021551

Practical data assimilation algorithms often contain hyper-parameters, which may arise due to, for instance, the use of certain auxiliary techniques like covariance inflation and localization in an ensemble Kalman filter, the re-parameterization of certain quantities such as model and/or observation error covariance matrices, and so on. Given the richness of the established assimilation algorithms, and the abundance of the approaches through which hyper-parameters are introduced to the assimilation algorithms, one may ask whether it is possible to develop a sound and generic method to efficiently choose various types of (sometimes high-dimensional) hyper-parameters. This work aims to explore a feasible, although likely partial, answer to this question. Our main idea is built upon the notion that a data assimilation algorithm with hyper-parameters can be considered as a parametric mapping that links a set of quantities of interest (e.g., model state variables and/or parameters) to a corresponding set of predicted observations in the observation space. As such, the choice of hyper-parameters can be recast as a parameter estimation problem, in which our objective is to tune the hyper-parameters in such a way that the resulted predicted observations can match the real observations to a good extent. From this perspective, we propose a hyper-parameter estimation workflow and investigate the performance of this workflow in an ensemble Kalman filter. In a series of experiments, we observe that the proposed workflow works efficiently even in the presence of a relatively large amount (up to 10^3) of hyper-parameters, and exhibits reasonably good and consistent performance under various conditions.

KEYWORDS

ensemble data assimilation, ensemble Kalman filter, iterative ensemble smoother, hyper-parameter optimization, correlation-based adaptive localization

1. Introduction

Data assimilation leverages the information contents of observational data to improve our understanding of quantities of interest (QoI), which could be model state variables and/or parameters, or their probability density functions (PDF) in a Bayesian estimation framework. Various challenges encountered in data assimilation problems lead to a rich list of assimilation algorithms developed from different perspectives, including, for instance, Kalman filter [1], extended Kalman filter [2], unscented Kalman filter [3], particle filter [4, 5], Gaussian sum filter [6], for sequential data assimilation problems; 3D- or 4-variational assimilation algorithms [7, 8]; and smoother algorithms for retrospective analysis [9].

To mitigate the computational costs in practical data assimilation problems, Monte Carlo or low-rank implementations of certain assimilation algorithms are investigated. Examples in this regard include ensemble Kalman filter (EnKF) and its variants [10–16], ensemble Kalman smoother [17], ensemble smoother [18], and their iterative versions [19–22], low-rank unscented Kalman filter [23, 24], ensemble or low-rank Gaussian sum filter [25–27].

In their practical forms, many assimilation algorithms may contain a certain number of hyper-parameters. Different from model parameters, hyper-parameters are variables that stem from assimilation algorithms and have influences on the assimilation results. As examples, one may consider the inflation factor and the length scale in covariance inflation and localization methods [13, 28–38], respectively, or parameters that are related to model error simulations or representations [39–42].

Often, a proper choice of algorithmic hyper-parameters is essential for obtaining a descent performance of data assimilation. With the presence of various mechanisms through which algorithmic hyper-parameters are introduced, in the literature there is a vast list of methods that are proposed to estimate hyper-parameters (while sometimes relying on empirical tuning). To the best of our knowledge, it appears that the current best practice is to focus on developing tailored estimation/tuning methods for individual mechanisms. With this observation, a natural question would be: Is it possible to develop a common method that can be employed to estimate different types of hyper-parameters associated with an assimilation algorithm?

This work can be considered as an attempt to find an affirmative answer to the above question. Our main idea here is to treat a data assimilation algorithm with hyper-parameters as a parametric mapping, which maps QoI (e.g., model state variables and/or parameters) to predicted observations in the observation space. From this perspective, it will be shown later that the choice of hyper-parameters can be converted to a nonlinear parameter estimation problem, which in turn can be solved through an iterative ensemble assimilation algorithm, similar to what have been done in the recent work of Luo [41] and Scheffler et al. [42]. Since ensemble-based data assimilation methods can be interpreted as some local gradient-based optimization algorithms [16, 43], we impose a restriction on the hyper-parameters under estimation, that is, they have to admit continuous values. In other words, we focus on the Continuous Hyper-parameter OPtimization (CHOP) problem, whereas tuning discrete hyper-parameters is beyond the scope of the current work.

It is worth mentioning that hyper-parameter optimization is a topic also often encountered in other research areas. For instance, in machine learning problems, there may exist various hyper-parameters (e.g., learning rate and batch size used in a training algorithm) that need to be optimized. Consequently, there are many techniques and tools developed in machine learning community to tackle hyper-parameter optimization problems [44–46]. Given the fact that data assimilation and machine learning problems bear certain differences [41], and the consideration that the focus of the current work is on developing an ensemble-based CHOP workflow for ensemble data assimilation algorithms, we do not introduce or compare hyper-parameter optimization techniques adopted in machine learning problems, although we do expect that hyper-parameter optimization techniques in machine learning community may also be extended to data assimilation problems.

In terms of novelty in the current work, to the best of our knowledge, CHOP appears to be the first ensemble-based hyper-parameter optimization workflow in the data assimilation community. As will be elaborated later, instead of producing a point estimation of hyper-parameters, the CHOP workflow generates an ensemble of such estimations. In doing so, a few practical advantages (similar to those pertaining to ensemble-based data assimilation algorithms) can be obtained, which include the ability of conducting uncertainty quantification and the derivative-free nature in the course of optimizing hyper-parameters. In addition to these practical advantages, CHOP can be seamlessly integrated into ensemble-based data assimilation algorithms to form a more automated assimilation workflow, which can automatically determine an ensemble of (near) optimal hyper-parameters with minimal manual interference, and possesses the capacity to simultaneously handle a large amount of hyper-parameters (a challenging issue seemingly not addressed by existing hyper-parameter optimization methods in the data assimilation community).

This work is organized as follows: We first formulate the CHOP problem, and propose a workflow (called CHOP workflow hereafter) to tackle the CHOP problem, which involves the use of an iterative ensemble smoother (IES) and a correlation-based adaptive localization scheme. We then investigate and report the performance of the CHOP workflow in a series of experiments. Finally, we conclude this study with some technical discussions and possible future works.

2. Problem statement and methodology

2.1. The CHOP problem

We illustrate the main idea behind the CHOP workflow in the setting of a sequential data assimilation problem, in which an EnKF is adopted with a certain number of hyper-parameters. Let $\mathbf{m} \in \mathbb{R}^m$ be an m-dimensional vector, which contains a set of model state variables and/or parameters. In the subsequent derivation of the solution to the CHOP problem, the dynamical system is not involved. As a result, we exclude the forecast step, and focus more on the analysis step, which applies an EnKF to update a background estimation $\mathbf{m}^b$ to the analysis $\mathbf{m}^a$.

Essentially, the EnKF can be treated as a parameterized vector mapping $\mathbf{f}_{\boldsymbol{\theta}} : \mathbf{m}^b \rightarrow \mathbf{m}^a$ that transforms $\mathbf{m}^b$ to $\mathbf{m}^a$, where $\boldsymbol{\theta}$ represents a set of algorithmic hyper-parameters to be estimated. In the context of data assimilation, the information contents of observational data, denoted by $\mathbf{d}^o \in \mathbb{R}^d$ in this work, are utilized for state and/or parameter update, whereas the update process also involves an observation operator, denoted by $\mathbf{h}$ here, which maps a background estimation $\mathbf{m}^b$ to some predicted data $\mathbf{h}\left(\mathbf{m}^b\right)$ in the observation space. We assume that the observations $\mathbf{d}^o$ contain some Gaussian white noise, which follows the normal distribution $N\left(\mathbf{0}, \mathbf{C}_d\right)$ with mean $\mathbf{0}$ and covariance $\mathbf{C}_d$. In addition, we denote the background ensemble by $\mathcal{M}^b \equiv \{\mathbf{m}_j^b\}_{j=1}^{N_e}$, and the analysis ensemble by $\mathcal{M}^a \equiv \{\mathbf{m}_j^a\}_{j=1}^{N_e}$, where j is the index of ensemble member, and N_e represents the number of ensemble members.

Under these settings, an analysis step of the EnKF can be represented as follows:

$$\begin{aligned}\mathbf{m}_j^a &= \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{m}_j^b, \mathcal{M}^b, \mathbf{d}_j^o, \mathbf{h}\right) \\ &\equiv \mathbf{f}\left(\boldsymbol{\theta}; \mathbf{m}_j^b, \mathcal{M}^b, \mathbf{d}_j^o, \mathbf{h}\right), \text{ for } j = 1, 2, \cdots, N_e. \qquad (1)\end{aligned}$$

In Equation (1), the concrete form of the mapping $\mathbf{f}$ will depend on the specific EnKF algorithm of choice. The quantities $\mathbf{m}_j^b$, $\mathcal{M}^b$, $\mathbf{d}_j^o$ and $\mathbf{h}$ are known, whereas the hyper-parameter vector $\boldsymbol{\theta}$ is to be estimated under a certain criterion, leading to a CHOP problem.

As an example, one may consider the case that an EnKF with perturbed observations is adopted, and covariance localization is introduced to the EnKF, such that the update formula is given as follows:

$$\begin{aligned}\mathbf{f}\left(\boldsymbol{\theta}; \mathbf{m}_j^b, \mathcal{M}^b, \mathbf{d}_j^o, \mathbf{h}\right) &= \mathbf{m}_j^b + \left(\mathbf{L}_{\boldsymbol{\theta}} \circ \mathbf{C}_m\right) \mathbf{h}^T \left(\mathbf{h}\left(\mathbf{L}_{\boldsymbol{\theta}} \circ \mathbf{C}_m\right) \mathbf{h}^T\right. \\ &\quad \left. + \ \mathbf{C}_d\right)^{-1} \left(\mathbf{d}_j^o - \mathbf{h}\mathbf{m}_j^b\right). \qquad (2)\end{aligned}$$

In Equation (2), we have assumed that $\mathbf{h}$ is a linear observation operator in this particular example, whereas $\mathbf{C}_m$ is the sample covariance matrix induced by the background ensemble $\mathcal{M}^b$; $\mathbf{L}_{\boldsymbol{\theta}}$ the localization matrix, which depends on some hyper-parameter(s) $\boldsymbol{\theta}$ (e.g., the length scale); and $\mathbf{L}_{\boldsymbol{\theta}} \circ \mathbf{C}_m$ stands for the Schur product of $\mathbf{L}_{\boldsymbol{\theta}}$ and $\mathbf{C}_m$. One insight from Equation (2) is that even $\mathbf{f}$ is a linear function of $\mathbf{m}_j^b$, in general $\mathbf{f}$ may have a nonlinear relation to the hyper-parameters $\boldsymbol{\theta}$.

2.2. Solution to the CHOP problem

In the current work, we treat CHOP as a parameter estimation problem, which can be solved through an ensemble-based, iterative assimilation algorithm, given the presence of nonlinearity in the CHOP problem. Specifically, we follow the idea in [22] to tackle the CHOP problem by minimizing the average of an ensemble of N_e cost functions $C_j^i\left(\boldsymbol{\theta}_j^i\right)$ at each iteration step (indexed by i):

$$\underset{\{\boldsymbol{\theta}_j^i\}_{j=1}^{N_e}}{\arg\min} \ \frac{1}{N_e} \sum_{j=1}^{N_e} C_j^i\left(\boldsymbol{\theta}_j^i\right), \qquad (3)$$

$$\begin{aligned}C_j^i\left(\boldsymbol{\theta}_j^i\right) &\equiv \frac{1}{2}\left\{\left(\mathbf{d}_j^o - \mathbf{g}\left(\boldsymbol{\theta}_j^i\right)\right)^T \mathbf{C}_{\mathbf{d}}^{-1} \left(\mathbf{d}_j^o - \mathbf{g}\left(\boldsymbol{\theta}_j^i\right)\right)\right. \\ &\left. + \gamma^{i-1}\left(\boldsymbol{\theta}_j^i - \boldsymbol{\theta}_j^{i-1}\right)^T \left(\mathbf{C}_{\boldsymbol{\theta}}^{i-1}\right)^{-1} \left(\boldsymbol{\theta}_j^i - \boldsymbol{\theta}_j^{i-1}\right)\right\}, \qquad (4)\end{aligned}$$

$$\mathbf{g}\left(\boldsymbol{\theta}_j^i\right) \equiv \mathbf{h}\left(\mathbf{m}_j^i\right) = \mathbf{h}\left(\mathbf{f}\left(\boldsymbol{\theta}_j^i; \mathbf{m}_j^b, \mathcal{M}^b, \mathbf{d}_j^o, \mathbf{h}\right)\right). \qquad (5)$$

In Equation (5), $\mathbf{g}\left(\boldsymbol{\theta}_j^i\right)$, equal to $\mathbf{h}\left(\mathbf{m}_j^i\right)$, corresponds to the predicted observations of $\mathbf{m}_j^i$, which in turn depends on the hyper-parameters $\boldsymbol{\theta}_j^i$ for a chosen assimilation algorithm $\mathbf{f}$. At the end of the iteration process, suppose that in total K iteration steps are executed to obtain $\boldsymbol{\theta}_j^K$, then we take $\mathbf{m}_j^a = \mathbf{m}_j^K = \mathbf{f}\left(\boldsymbol{\theta}_j^K; \mathbf{m}_j^b, \mathcal{M}^b, \mathbf{d}_j^o, \mathbf{h}\right)$, $\forall j = 1, 2, \cdots, N_e$.

As implied in Equations (3) and (4), the main idea behind the proposed CHOP workflow is to find, at each iteration step, an ensemble of hyper-parameters $\boldsymbol{\Theta}^i \equiv \left\{\boldsymbol{\theta}_j^i\right\}_{j=1}^{N_e}$ that renders lower average data mismatch, in terms of

$$\sum_{j=1}^{N_e} \left(\mathbf{d}_j^o - \mathbf{g}\left(\boldsymbol{\theta}_j^i\right)\right)^T \mathbf{C}_{\mathbf{d}}^{-1} \left(\mathbf{d}_j^o - \mathbf{g}\left(\boldsymbol{\theta}_j^i\right)\right) / N_e,$$

than the previous ensemble $\boldsymbol{\Theta}^{i-1}$ does. However, as in many ill-posed inverse problems, it is desirable to avoid over-fitting the observations. To this end, a regularization term, in the form of $\left(\boldsymbol{\theta}_j^i - \boldsymbol{\theta}_j^{i-1}\right)^T \left(\mathbf{C}_{\boldsymbol{\theta}}^{i-1}\right)^{-1} \left(\boldsymbol{\theta}_j^i - \boldsymbol{\theta}_j^{i-1}\right)$, is introduced into the cost function $C_j^i\left(\boldsymbol{\theta}_j^i\right)$ in Equation (4), whereas $\mathbf{C}_{\boldsymbol{\theta}}^{i-1}$ corresponds to the sample covariance matrix induced by the ensemble of hyper-parameters $\boldsymbol{\Theta}^{i-1} = \left\{\boldsymbol{\theta}_j^{i-1}\right\}_{j=1}^{N_e}$, and can be expressed as $\mathbf{C}_{\boldsymbol{\theta}}^{i-1} = \mathbf{S}_\theta^{i-1}(\mathbf{S}_\theta^{i-1})^T$, with $\mathbf{S}_\theta^{i-1}$ being a square root matrix defined in Equation (9) later. The positive scalar γ^{i-1} can be considered a coefficient that determines the relative weight between the data mismatch and the regularization terms at each iteration step, and we will discuss its choice later.

Another implication from Equations (3) to (5) is that instead of rendering a single estimation of the hyper-parameters, we provide an ensemble of such estimates, and each of them (e.g., $\boldsymbol{\theta}_j^i$) is associated with a model state and/or parameter vector

$\mathbf{m}_j^i$. The presence of multiple estimates $\boldsymbol{\theta}_j^i$ not only provides the possibility of uncertainty analysis in a CHOP problem, but also avoids the need to explicitly evaluate the gradients of $\mathbf{g}$ with respect to $\boldsymbol{\theta}_j^i$ in the course of solving the minimization problem in Equation (3).

Equation (3) can be approximately solved by an IES, given as follows [22]:

$$\boldsymbol{\theta}_j^i = \boldsymbol{\theta}_j^{i-1} + \mathbf{K}^{i-1}\left(\mathbf{d}_j^o - \mathbf{g}\left(\boldsymbol{\theta}_j^{i-1}\right)\right), \, j = 1, 2, \cdots, N_e; \quad (6)$$

$$\mathbf{K}^{i-1} \equiv \mathbf{S}_\theta^{i-1}(\mathbf{S}_\mathbf{g}^{i-1})^T \left(\mathbf{S}_\mathbf{g}^{i-1}(\mathbf{S}_\mathbf{g}^{i-1})^T + \gamma^{i-1}\mathbf{C}_d\right)^{-1}; \quad (7)$$

$$\bar{\boldsymbol{\theta}}^{i-1} \equiv \frac{1}{N_e}\sum_{j=1}^{N_e}\boldsymbol{\theta}_j^{i-1}; \quad (8)$$

$$\mathbf{S}_\theta^{i-1} \equiv \frac{1}{\sqrt{N_e - 1}}\left[\boldsymbol{\theta}_1^{i-1} - \bar{\boldsymbol{\theta}}^{i-1}, \boldsymbol{\theta}_2^{i-1} - \bar{\boldsymbol{\theta}}^{i-1}, \cdots, \boldsymbol{\theta}_{N_e}^{i-1} - \bar{\boldsymbol{\theta}}^{i-1}\right]; (9)$$

$$\mathbf{S}_\mathbf{g}^{i-1} \equiv \frac{1}{\sqrt{N_e - 1}}\left[\mathbf{g}\left(\boldsymbol{\theta}_1^{i-1}\right) - \mathbf{g}\left(\bar{\boldsymbol{\theta}}^{i-1}\right), \mathbf{g}\left(\boldsymbol{\theta}_2^{i-1}\right) - \mathbf{g}\left(\bar{\boldsymbol{\theta}}^{i-1}\right), \cdots, \mathbf{g}\left(\boldsymbol{\theta}_{N_e}^{i-1}\right) - \mathbf{g}\left(\bar{\boldsymbol{\theta}}^{i-1}\right)\right]. \quad (10)$$

As one of the attractive properties of various ensemble-based assimilation algorithms, this iteration process does not explicitly involve the gradients of $\mathbf{g}$, $\mathbf{h}$ (the observation operator) or $\mathbf{f}$ (the assimilation algorithm) with respect to the hyper-parameters $\boldsymbol{\theta}$, which helps to reduce the complexities of implementing the IES algorithm.

In a practical implementation, the update formulas from Equations (6) to (7) are re-written as follows:

$$\boldsymbol{\theta}_j^i = \boldsymbol{\theta}_j^{i-1} + \mathbf{S}_\theta^{i-1}(\tilde{\mathbf{S}}_\mathbf{g}^{i-1})^T\left(\tilde{\mathbf{S}}_\mathbf{g}^{i-1}(\tilde{\mathbf{S}}_\mathbf{g}^{i-1})^T + \gamma^{i-1}\mathbf{I}_d\right)^{-1} \left(\tilde{\mathbf{d}}_j^o - \tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)\right); \quad (11)$$

$$\tilde{\mathbf{S}}_\mathbf{g}^{i-1} \equiv \mathbf{C}_d^{-1/2}\mathbf{S}_\mathbf{g}^{i-1}; \; \tilde{\mathbf{d}}_j^o \equiv \mathbf{C}_d^{-1/2}\mathbf{d}_j^o; \; \tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right) \equiv \mathbf{C}_d^{-1/2}\mathbf{g}\left(\boldsymbol{\theta}_j^{i-1}\right). \quad (12)$$

In Equation (11), $\mathbf{I}_d$ represents the d-dimensional identity matrix. In Equation (12), the quantities $\mathbf{S}_\mathbf{g}^{i-1}$, $\mathbf{d}_j^o$ and $\mathbf{g}\left(\boldsymbol{\theta}_j^{i-1}\right)$ in the observation space are normalized by a square root $\mathbf{C}_d^{-1/2}$ of the observation error covariance matrix. After this normalization, a singular value decomposition (SVD) is applied to $\tilde{\mathbf{S}}_\mathbf{g}^{i-1}$, while avoiding the potential issue of different magnitudes of observations when forming the square root matrix $\mathbf{S}_\mathbf{g}^{i-1}$. Suppose that through the SVD, we have

$$\tilde{\mathbf{S}}_\mathbf{g}^{i-1} = \tilde{\mathbf{U}}^{i-1}\tilde{\boldsymbol{\Sigma}}^{i-1}\left(\tilde{\mathbf{V}}^{i-1}\right)^T. \quad (13)$$

To strengthen the numerical stability of the IES algorithm, we discard a number of relatively small singular values, which results in a truncated SVD such that

$$\tilde{\mathbf{S}}_\mathbf{g}^{i-1} \approx \hat{\mathbf{U}}^{i-1}\hat{\boldsymbol{\Sigma}}^{i-1}\left(\hat{\mathbf{V}}^{i-1}\right)^T. \quad (14)$$

The truncation criterion adopted in the current work is as follows: Suppose that the matrix $\tilde{\boldsymbol{\Sigma}}^{i-1}$ contains a number of R singular values $\tilde{\sigma}_1^{i-1}, \tilde{\sigma}_2^{i-1}, \cdots, \tilde{\sigma}_R^{i-1}$ arranged in the descending order, then we keep the first r leading singular values such that $\sum_{\ell=1}^{r}\tilde{\sigma}_\ell^{i-1}/\sum_{\ell=1}^{R}\tilde{\sigma}_\ell^{i-1} \le 99\%$ and $\sum_{\ell=1}^{r+1}\tilde{\sigma}_\ell^{i-1}/\sum_{\ell=1}^{R}\tilde{\sigma}_\ell^{i-1} > 99\%$. In Equation (14), the matrix $\hat{\boldsymbol{\Sigma}}^{i-1}$ takes the leading singular values $\tilde{\sigma}_1^{i-1}, \tilde{\sigma}_2^{i-1}, \cdots, \tilde{\sigma}_r^{i-1}$ as its diagonal elements. Accordingly, the matrices $\hat{\mathbf{U}}^{i-1}$ and $\hat{\mathbf{V}}^{i-1}$ consist of eigen-vectors that correspond to these kept leading singular values.

Inserting Equation (14) into Equation (11), one obtains a modified update formula:

$$\boldsymbol{\theta}_j^i \approx \boldsymbol{\theta}_j^{i-1} + \mathbf{S}_\theta^{i-1}\hat{\mathbf{V}}^{i-1}\hat{\boldsymbol{\Sigma}}^{i-1}\left(\left(\hat{\boldsymbol{\Sigma}}^{i-1}\right)^2 + \gamma^{i-1}\mathbf{I}_r\right)^{-1} \left(\hat{\mathbf{U}}^{i-1}\right)^T\left(\tilde{\mathbf{d}}_j^o - \tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)\right), \quad (15)$$

which is used in all numerical experiments later. In Equation (15), $\left(\hat{\boldsymbol{\Sigma}}^{i-1}\right)^2 \equiv \hat{\boldsymbol{\Sigma}}^{i-1}\hat{\boldsymbol{\Sigma}}^{i-1}$, and $\mathbf{I}_r$ stands for the r-dimensional identity matrix.

As mentioned previously, γ^{i-1} can be considered as a coefficient that determines the relative weight between the data mismatch and regularization terms. In the update formula, e.g., Equation (11) or (15), one can see that in effect, γ^{i-1} affects the change $\boldsymbol{\theta}_j^i - \boldsymbol{\theta}_j^{i-1}$ of the hyper-parameters, which is also referred to as the step size of the iteration hereafter. Following the discussions in [22, 43], it can be shown that the update formula, Equation (11) or (15), is derived by implicitly linearizing $\mathbf{g}\left(\boldsymbol{\theta}_j^i\right), \forall j = 1, 2, \cdots, N_e$, around the ensemble mean $\bar{\boldsymbol{\theta}}^{i-1}$ (through the first-order Taylor approximation) at each iteration step[1]. In this regard, an implication is that the step size cannot be too big in order to make the linearization strategy approximately valid. On the other hand, a too small step size will slow down the convergence of the iteration process. As a result, in our implementation of the IES algorithm (e.g., Equation 11), we choose γ^{i-1} in such a way that the influences of the two terms, $\tilde{\mathbf{S}}_\mathbf{g}^{i-1}(\tilde{\mathbf{S}}_\mathbf{g}^{i-1})^T$ and $\gamma^{i-1}\mathbf{I}_d$ are comparable (in contrast to the choice that one term dominates the other). Here, the influence is measured in terms of the trace of the respective term. As a consequence of this notion, we have $\gamma^{i-1} = \alpha^{i-1}\,\text{trace}\left(\tilde{\mathbf{S}}_\mathbf{g}^{i-1}(\tilde{\mathbf{S}}_\mathbf{g}^{i-1})^T\right)/\,\text{trace}\left(\mathbf{I}_d\right) = \alpha^{i-1}\,\text{trace}\left(\tilde{\mathbf{S}}_\mathbf{g}^{i-1}(\tilde{\mathbf{S}}_\mathbf{g}^{i-1})^T\right)/d$, where $\alpha^{i-1} > 0$ is the actual coefficient to be tuned.

When the truncated SVD is applied to $\tilde{\mathbf{S}}_\mathbf{g}^{i-1}$, the choice of γ^{i-1} for Equation (15) boils

[1] By "implicitly linearizing" we mean that the derivation of the update formula adopts the concept of linearization, but there is no need to actually evaluate the gradients of $\mathbf{g}$ with respect to $\bar{\boldsymbol{\theta}}^{i-1}$.

down to

$$\gamma^{i-1} = \alpha^{i-1}\,\text{trace}\left(\left(\hat{\boldsymbol{\Sigma}}^{i-1}\right)^2\right) / \,\text{trace}\,(\mathbf{I}_r)$$
$$= \alpha^{i-1} \sum_{\ell=1}^{r} \left(\tilde{\sigma}_\ell^{i-1}\right)^2 / r, \qquad (16)$$

At the beginning of the iteration, we let $\alpha^0 = 1$. Subsequently, We use a backtrack line search strategy similar to that in [21] to tune the coefficient value. Specifically, if the average data mismatch at step i is lower than that at step $(i-1)$, then we accept the estimated hyper-parameters $\boldsymbol{\theta}_j^i$, and move to the next iteration step. To this end, we reduce the coefficient value by setting $\alpha^i = 0.9\alpha^{i-1}$, which aims to help increase the step size at the next iteration step, similar to the idea behind the trust-region algorithm [47].

On the other hand, if the average data mismatch value at step i becomes higher than that at step $(i-1)$, then the estimated hyper-parameters $\boldsymbol{\theta}_j^i$ are not used for the next iteration step. Instead, a few attempts (say K_{trial}) are conducted to search for better estimations, leading to a so-called inner-loop iteration (if any), which is adopted for a distinction from the upper-level iteration process (called outer-loop iteration). These are done by doubling the coefficient value $\alpha_s^{i-1} = 2\alpha_{s-1}^{i-1}, s = 1, 2, \cdots, K_{trial}$, for each trial, with $\alpha_0^{i-1} = \alpha^{i-1}$, and then re-running the update formula (Equation 15) with a new γ^{i-1} value calculated by Equation (16), wherein the modified α_s^{i-1} value is adopted for the calculation. This strategy is again similar to the setting of the trust-region algorithm, and is also in line with the analysis in [22], where it is shown that as long as the linearization strategy is approximately valid, the data mismatch values tend to decrease over the iteration steps. As such, it is sensible to increase the coefficient value (hence shrink the step size), as this helps to improve the accuracy of the first-order Taylor approximation (hence the validity of the linearization strategy). The trial process will be terminated if an average data mismatch value (obtained by using an enlarged coefficient value α_s^{i-1}) is found lower than that at the iteration step $(i-1)$, or if the maximum trial number (set to 5) is reached. At the end of the trial process, we set $\alpha^i = \alpha_{K_{trial}}^{i-1}$, and take $\boldsymbol{\theta}_j^i$ as those obtained from the last trial step.

An additional aspect of the IES algorithm is the stopping criteria. Three such criteria are adopted in the outer-loop iteration process, which include: (1) the maximum iteration step, which is set to be 10; (2) the threshold for the relative change of the average data mismatch values at two consecutive iteration steps, which is set to be 0.01%; (3) the threshold for the average data mismatch value, which is set to be $4 \times \#(\mathbf{d}^o)$ (four times the number of observations, with $\#(\mathbf{d}^o)$ being the number of elements in $\mathbf{d}^o$). In other words, the iteration process will stop if the maximum iteration step is reached. Additionally, the iteration process will also stop if the relative change of the average data mismatch values at two consecutive iteration steps, or the average data mismatch value itself at a certain iteration step, is less than their respective threshold value.

In terms of computational cost, the original analysis scheme (e.g., Equation 1), applies the update formula only once. In contrast, in a CHOP problem, one needs to apply the update formula multiple times during the iteration process. As such, it becomes computationally more expensive to solve the CHOP problem than a straightforward application of the EnKF analysis scheme (if one ignores the potential cost of searching for proper hyper-parameter values). In practical problems, however, the computationally most expensive part of an assimilation workflow often lies in running the dynamical system (i.e., at the forecast step), whereas it is computationally much cheaper to execute the analysis step. Within this context, it is expected that solving the CHOP problem will only lead to a negligible (hence affordable) overhead of computational cost to the whole assimilation workflow.

2.3. Localization in the CHOP problem

In many data assimilation problems, the heavy cost of running the dynamical system also puts a constraint on how many ensemble members one can afford to use. Often, a trade-off has to be made so that one employs an ensemble data assimilation algorithm with a relatively small ensemble size for runtime reduction. One consequence of this limited ensemble size is that there could be substantial sampling errors when using the statistics (e.g., covariance and correlation) estimated from the small ensemble in the update formula. In addition, rank deficiencies of estimated covariance matrices would also take place. These noticed issues often lead to degraded performance of data assimilation. To mitigate the impacts of sampling errors and rank deficiency, localization techniques (e.g., [13, 30–32, 48, 49]), are often employed.

In the CHOP problem, we note that localization is conducted with respect to hyper-parameters (e.g., in Equations 11 or 15), in spite of the possible presence of another localization scheme adopted in the assimilation algorithm (e.g., as in Equation 2).

Many localization methods are based on the distances between the physical locations of certain pairs of quantities, which can be either pairs of two model variables as in model-space localization schemes (e.g., [13]), or pairs of one model variable and one observation as in observation-space localization schemes (se.g., [49]). In the CHOP problem, however, in certain circumstances it may be challenging to apply distance-based localization, as in the update formula, Equations (11) or (15), certain hyper-parameters may not possess clearly defined physical locations, so that the concept of physical distance itself may not be valid.

To circumvent this difficulty, we adopt a correlation-based adaptive localization scheme proposed in [50]. For illustration, without loss of generality, suppose that when localization is not adopted, the update formula is in the form of

$$\boldsymbol{\theta}_j^i = \boldsymbol{\theta}_j^{i-1} + \tilde{\mathbf{K}}^{i-1}\left(\tilde{\mathbf{d}}_j^o - \tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)\right), \tag{17}$$

where $\tilde{\mathbf{K}}^{i-1}$ is a Kalman-gain-like matrix and $\left(\tilde{\mathbf{d}}_j^o - \tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)\right)$ the corresponding innovation term. With the presence of localization, then the update formula is modified as

$$\boldsymbol{\theta}_j^i = \boldsymbol{\theta}_j^{i-1} + \left(\mathbf{L}^{i-1} \circ \tilde{\mathbf{K}}^{i-1}\right)\left(\tilde{\mathbf{d}}_j^o - \tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)\right), \tag{18}$$

where $\mathbf{L}^{i-1}$ is a $h \times d$ localization matrix to be constructed, with h and d being the vector lengths of $\boldsymbol{\theta}_j^i$ and $\tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)$ (or $\tilde{\mathbf{d}}_j^o$), respectively. In Equation (18), the localization scheme is similar to observation-space localization, but the localization matrix $\mathbf{L}^{i-1}$ acts on the Kalman-gain-like matrix $\tilde{\mathbf{K}}^{i-1}$.

The construction of the localization matrix $\mathbf{L}^{i-1}$ is based on the notion of causality detection between the hyper-parameters $\boldsymbol{\theta}_j^i$ and the predicted observations $\tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)$ [50]. To see the rationale behind this notion, let $\tilde{\mathbf{d}}_j^{i-1,pred} \equiv \tilde{\mathbf{g}}\left(\boldsymbol{\theta}_j^{i-1}\right)$ and $\Delta\tilde{\mathbf{d}}_j^{i-1} \equiv \tilde{\mathbf{d}}_j^o - \tilde{\mathbf{d}}_j^{i-1,pred}$, and re-write Equation (18) into an equivalent, element-wise form

$$\theta_{j,s}^i = \theta_{j,s}^{i-1} + \sum_{t=1}^{d}\left(L_{s,t}^{i-1}\tilde{K}_{s,t}^{i-1}\right)\Delta\tilde{d}_{j,t}^{i-1}, \text{ for } s = 1, 2, \cdots, h, \tag{19}$$

where $\theta_{j,s}^i$, $\theta_{j,s}^{i-1}$ and $\Delta\tilde{d}_{j,t}^{i-1}$ represent the $s-$th or the $t-$th element of $\boldsymbol{\theta}_j^i$, $\boldsymbol{\theta}_j^{i-1}$ and $\Delta\tilde{\mathbf{d}}_j^{i-1}$, respectively; while $L_{s,t}^{i-1} \in [0, 1]$ and $\tilde{K}_{s,t}^{i-1}$ stand for the elements on the $s-$th row and the $t-$th column of the matrices $\mathbf{L}^{i-1}$ and $\tilde{\mathbf{K}}^{i-1}$, respectively.

The implication of Equation (19) is that the innovation elements $\Delta\tilde{d}_{j,t}^{i-1}$ ($t = 1, 2, \cdots, d$) contribute to the change $\theta_{j,s}^i - \theta_{j,s}^{i-1}$ of the $s-$th hyper-parameter, and the degree of the contribution of each innovation element $\Delta\tilde{d}_{j,t}^{i-1}$ is determined by the element $\tilde{K}_{s,t}^{i-1}$ (if no localization), together with the tapering coefficient $L_{s,t}^{i-1}$ (if with localization).

In the notion of causality detection to choose the value of $L_{s,t}^{i-1}$, the main idea is that if there is a causality from the $s-$th element of hyper-parameters to the $t-$th element of innovations, then $\Delta\tilde{d}_{j,t}^{i-1}$ should be used for updating $\theta_{j,s}^{i-1}$ to $\theta_{j,s}^i$, meaning that $L_{s,t}^{i-1} \neq 0$. In contrast, if there is no causality therein, then it is sensible to exclude $\Delta\tilde{d}_{j,t}^{i-1}$ so that it makes no contribution to the update of $\theta_{j,s}^{i-1}$ to $\theta_{j,s}^i$, meaning that $L_{s,t}^{i-1} = 0$.

Here, the statistics used to measure the causality is the sample cross correlations (e.g., denoted by $\rho_{s,t}^{i-1}$) between the elements of an ensemble of hyper-parameters (e.g., $\theta_{j,s}^{i-1}$ for $j = 1, 2, \cdots, N_e$) and the corresponding ensemble of innovations (e.g., $\Delta\tilde{d}_{j,t}^{i-1}$ for $j = 1, 2, \cdots, N_e$). Intuitively, when the magnitude of a sample correlation, say $\rho_{s,t}^{i-1}$, is relatively high (e.g., close to 1), then one tends to believe that there is a true causality from the $s-$th element of hyper-parameters to the $t-$th element of innovations. On the other hand, when the magnitude of $\rho_{s,t}^{i-1}$ is relatively low (e.g., close, but not exactly equal, to 0), then more caution is needed. This is because when a limited ensemble size N_e is adopted, the induced sampling errors can cause spurious correlations, such that even there is no causality between a hyper-parameter and an innovation element, the estimated sample correlation may not be identical to zero.

Taking into account the above consideration, we assign values to $L_{s,t}^{i-1}$ following a method in [50]:

$$L_{s,t}^{i-1} = f_{GC}\left(\frac{1 - |\rho_{s,t}^{i-1}|}{1 - 3/\sqrt{N_e}}\right), N_e > 9, \tag{20}$$

where f_{GC} is the Gaspari-Cohn (GC) function [51], which, for a scalar input $z \geq 0$, satisfies

$$f_{GC}(z) = \begin{cases} -\frac{1}{4}z^5 + \frac{1}{2}z^3 + \frac{5}{8}z^3 - \frac{5}{3}z^2 + 1\,, & \text{if } 0 \leq z \leq 1\,; \\ \frac{1}{12}z^5 - \frac{1}{2}z^4 + \frac{5}{8}z^3 + \frac{5}{3}z^2 - 5z + 4 - \frac{2}{3}z^{-1}\,, & \text{if } 1 < z \leq 2\,; \\ 0\,, & \text{if } z > 2\,. \end{cases} \tag{21}$$

Note that in general, choosing Equation (21) as the tapering function may not be optimal, and other types of tapering functions may also be considered, see, for instance, [52].

In Equation (20), the factor $3/\sqrt{N_e}$ is adopted for the following reason: When the true correlation between the s-th hyper-parameter and the t-th innovation is 0, but the sample correlation is evaluated with a sample size of N_e, then the sampling errors follow a Gaussian distribution $N(0, 1/N_e)$ asymptotically, see [50] and the reference therein. Therefore, under the hypothesis (denoted by H_0 hereafter) that the true correlation is 0, we compare the magnitude of the sample correlation $\rho_{s,t}^{i-1}$ with three times the standard deviation (STD) ($3/\sqrt{N_e}$). The larger $|\rho_{s,t}^{i-1}|$ is, the more confident we are that H_0 should be rejected, meaning it is more likely that there is a true (non-zero) correlation between the s-th hyper-parameter and the t-th innovation. As such, $L_{s,t}^{i-1}$ will receive a larger value. On the other hand, the value of $L_{s,t}^{i-1}$ becomes smaller as $|\rho_{s,t}^{i-1}|$ decreases.

In comparison to distance-based localization, a few additional benefits of the above correlation-based localization include: better abilities to hand non-local observations, time-lapse effects of observations and big observation datasets; and improved adaptivity to different types of model parameters/state variables. For more details, readers are referred to [50].

3. Numerical results

The L96 model [53] is taken as the testbed in the current study. For a N_L-dimensional L96 model, its dynamic behavior is described by the following ordinary differential equations (ODEs):

$$\frac{dx_e}{dt} = \left(x_{e+1} - x_{e-2}\right) x_{e-1} - x_e + F, \; e = 1, \cdots, N_L. \tag{22}$$

For consistency, $x_{-1} = x_{N_L - 1}$, $x_0 = x_{N_L}$ and $x_1 = x_{N_L+1}$ in Equation (22). The driving force term F is set to 8 throughout this work. The L96 model is integrated forward in time by the fourth-order Runge-Kutta method with a constant integration step of 0.05 time units (dimensionless).

Similar to the idea of cross-validating the reliability and performance of a machine learning model through certain statistical tests [54], in the experiments below, a few statistics are adopted to characterize the performance of data assimilation. These include the root mean square error (RMSE) E_m, ensemble spread S_{en} and data mismatch E_d. As will be seen below, RMSE computes a normalized Euclidean distance between an estimate and the ground truth in the model space, whereas data mismatch calculates a similar distance between predicted and real observations in the observation space. On the other hand, ensemble spread provides a measure of ensemble variability.

To compute these statistics, let $\mathbf{m}$ be a m-dimensional vector of estimated model state variables and/ or parameters that are of interest, $\mathbf{d}^{pred} \equiv \mathbf{h}(\mathbf{m})$ the corresponding predicted observation, with $\mathbf{h}$ being the observation operator, then given the reference $\mathbf{m}^{ref}$ (ground truth), we define the RMSE of $\mathbf{m}$ as

$$E_m = \|\mathbf{m} - \mathbf{m}^{ref}\|_2 / \sqrt{m}, \tag{23}$$

where the operator $\| \bullet \|_2$ returns the Euclidean norm of its operand $\bullet$.

In addition, assume that the real observation is $\mathbf{d}^o$, which is contaminated by some zero-mean Gaussian white noise, and is associated with an observation error covariance matrix $\mathbf{C}_d$, then we define the data mismatch of $\mathbf{m}$ as

$$E_d = \left(\mathbf{d}^o - \mathbf{d}^{pred}\right)^T \mathbf{C}_d^{-1} \left(\mathbf{d}^o - \mathbf{d}^{pred}\right). \tag{24}$$

For the definition of ensemble spread, let $\mathcal{M} = \left\{\mathbf{m}_j \equiv \left[m_{j,1}, m_{j,2}, \cdots m_{j,m}\right]^T\right\}_{j=1}^{N_e}$ be an ensemble of estimated model state variables/parameters, where $m_{j,k}$ denotes the k-th element of $\mathbf{m}_j$ ($k = 1, 2, \cdots, m$). Based on $\mathcal{M}$, we construct a vector $\mathbf{S} \equiv [\sigma_1, \sigma_2, \cdots, \sigma_m]^T$, where σ_k denotes the sample standard deviation with respect to the ensemble $\{m_{j,k}\}_{j=1}^{N_e}$, and compute the ensemble spread as

$$S_{en} = \|\mathbf{S}\|_2 / \sqrt{m}. \tag{25}$$

3.1. Experiments in a 40-dimensional L96 system

3.1.1. Experiment settings

We start from the common choice of $N_L = 40$ in the literature, while considering a much larger N_L value later on. We run the L96 model from time 0 to time 5,000 (which corresponds to 100,000 integration steps in total), and compute the long-term (lt) temporal mean $\hat{\mathbf{m}}^{lt}$ and covariance $\hat{\mathbf{C}}^{lt}$ based on the model variables at all integration steps.

In each of the experiments below, we draw a random sample from the Gaussian distribution $N\left(\hat{\mathbf{m}}^{lt}, \hat{\mathbf{C}}^{lt}\right)$, and use this sample as the initial condition to start the simulation of the L96 model in a transition time window of 250 time units (corresponding to 5,000 integration steps).

The model variables obtained at the end of the transition time window is then taken as the initial values to simulate reference model variables in an assimilation time window of 250 time units. Data assimilation is conducted within this assimilation time window to estimate reference model variables at different time steps, based on a background ensemble of model variables and noisy observations that are related to reference model variables through a certain observation system. The initial background ensemble (at the first time instance of the assimilation time window) is generated by drawing a specified number N_e of samples from the Gaussian distribution $N\left(\hat{\mathbf{m}}^{lt}, \hat{\mathbf{C}}^{lt}\right)$. The ensemble size N_e may change with the experiments, as will be specified later.

For a generic vector $\mathbf{m}$ of model state variables/parameters, the observation system adopted in the experiments is linear and in the form of

$$\begin{aligned} \mathbf{d} &= \mathbf{H}\mathbf{m} \\ &= \left[m_1, m_{1+\Delta n}, m_{1+2\Delta n}, \cdots, m_{1+M\Delta n}\right]^T, \end{aligned} \tag{26}$$

where $\mathbf{H}$ is a matrix extracting elements $m_1, m_{1+\Delta n}, m_{1+2\Delta n}, \cdots$ from $\mathbf{m}$, the integer Δn represents an increment of model-variable index, and M is the largest integer such that $1 + M\Delta n \le N_L$. The value of Δn may also vary in different experiments. As such, its concrete value will be mentioned in individual experiments later. For convenience, hereafter we may also use the shorthand notation $\{1 : \Delta n : N_L\}$ to denote the set $\{1, 1 + \Delta n, 1 + 2\Delta n, \cdots\}$ of indices. Similar notations will also be used elsewhere later.

In the experiments, we assume that the observation operator $\mathbf{H}$ is perfect and known to us. When applying Equation (26) to reference model variables to generate real observations for data assimilation, we add to the outputs of Equation (26) some Gaussian white noise $\boldsymbol{\epsilon}$, which is assumed to follow the Gaussian distribution $N(\mathbf{0}_{M+1}, \mathbf{I}_{M+1})$, with $\mathbf{0}_{M+1}$ and $\mathbf{I}_{M+1}$ being the $(M+1)$-dimensional zero vector, and the $(M+1)$-dimensional identity matrix, respectively. The frequency for us to collect the measurements is every Δt integration steps, whose value will also be specified in respective experiments.

The base assimilation algorithm adopted here is the EnKF with perturbed observations [55], in which the update formula reads:

$$\mathbf{m}_j^a = \mathbf{m}_j^b + \mathbf{C}_m \mathbf{H}^T \left(\mathbf{H} \mathbf{C}_m \mathbf{H}^T + \mathbf{C}_d \right)^{-1} \left(\mathbf{d}_j^o - \mathbf{H} \mathbf{m}_j^b \right), \\ \text{for } j = 1, 2, \cdots, N_e, \tag{27}$$

where $\mathbf{C}_m$ is the sample covariance matrix of the background ensemble $\mathcal{M}^b \equiv \{\mathbf{m}_j^b\}_{j=1}^{N_e}$, and $\mathbf{d}_j^o$ stands for perturbations with respect to the real observation $\mathbf{d}^o$.

Covariance inflation and localization are then introduced to Equation (27) to strengthen the performance of the EnKF. We note that our purpose here is to demonstrate how the CHOP workflow can be implemented on top of certain chosen inflation and localization techniques, yet the CHOP workflow itself cannot be used to design new inflation or localization techniques.

Specifically, in this study, covariance inflation is conducted on the background ensemble, in such a way that $\mathcal{M}^b$ is replaced by a modified background ensemble $\tilde{\mathcal{M}}^b \equiv \{\tilde{\mathbf{m}}_j^b\}_{j=1}^{N_e}$ with $\tilde{\mathbf{m}}_j^b = \bar{\mathbf{m}}^b + (1+\delta) \left(\mathbf{m}_j^b - \bar{\mathbf{m}}^b \right)$, where $\bar{\mathbf{m}}^b$ is the ensemble mean of the members in $\mathcal{M}^b$, and $\delta \geq 0$ is the inflation factor to be determined through a certain criterion. Accordingly, the sample covariance $\mathbf{C}_m$ in Equation (27) should be replaced by $\tilde{\mathbf{C}}_m = (1+\delta)^2 \mathbf{C}_m$, which is larger than $\mathbf{C}_m$ (hence the name covariance inflation).

On the other hand, localization is implemented by replacing the Kalman gain matrix $\tilde{\mathbf{K}} = \tilde{\mathbf{C}}_m \mathbf{H}^T \left(\mathbf{H} \tilde{\mathbf{C}}_m \mathbf{H}^T + \mathbf{C}_d \right)^{-1}$ by the Schur product $\mathbf{L} \circ \tilde{\mathbf{K}}$, where $\mathbf{L}$ is the localization matrix, whose element, say, $L_{s,t}$ on the s-th row and the t-th column of $\mathbf{L}$, is determined by the "physical" distance between the s-th model variable m_s and the t-th observation element d_t. For the observation system in Equation (26), d_t corresponds to the observation at the model-variable location $o = (1 + (t-1)\Delta n)$ (in terms of model-variable index). As such, the element $L_{s,t}$ is computed as follows:

$$L_{s,t} = f_{GC} \left(\frac{dist_{s,t}}{\lambda} \right), \tag{28}$$

$$dist_{s,t} = \min \left(|s - o|/N_L, 1 - |s - o|/N_L \right). \tag{29}$$

In Equation (28), f_{GC} is the Gaspari-Cohn function (see Eq. 21), $dist_{s,t}$ represents a normalized distance between the s-th model variable and the t-th observation element (which is located on the o-th model grid/index), and λ is the length scale, whose value is chosen under a certain criterion. Equation (29) computes the distance between the t-th and o-th model grids/indices, which is normalized by the total number N_L of the model grids (equal to the dimension of the L96 model in this case). Note that $dist_{s,t}$ takes the minimum value between $|s - o|/N_L$ and $1 - |s - o|/N_L$, due to the circular nature of the L96 model. In the sequel, we re-write $\mathbf{L}$ as $\mathbf{L}(\lambda)$ to indicate the dependence of $\mathbf{L}$ on λ.

Taking into account the presence of both covariance inflation and localization, the base assimilation algorithm, Equation (27), is modified as follows:

$$\mathbf{m}_j^a = \left[\bar{\mathbf{m}}^b + (1+\delta) \left(\mathbf{m}_j^b - \bar{\mathbf{m}}^b \right) \right] \\ + \left\{ \mathbf{L}(\lambda) \circ \left[\mathbf{C}_m \mathbf{H}^T \left(\mathbf{H} \mathbf{C}_m \mathbf{H}^T + \mathbf{C}_d / (1+\delta)^2 \right)^{-1} \right] \right\} \\ \left(\mathbf{d}_j^o - \mathbf{H} \left[\bar{\mathbf{m}}^b + (1+\delta) \left(\mathbf{m}_j^b - \bar{\mathbf{m}}^b \right) \right] \right). \tag{30}$$

The update formula in Equation (30) thus contains two hyper-parameters, the inflation factor δ and the length scale λ. With the known background ensemble $\mathcal{M}^b$ (hence $\mathbf{m}_j^b$, $\bar{\mathbf{m}}^b$, and $\mathbf{C}_m$) and the quantities $\mathbf{d}_j^o$, $\mathbf{C}_d$, and $\mathbf{H}$, the relation between the analysis $\mathbf{m}_j^a$ and the hyper-parameters is complex (and nonlinear in general), even with a rather simple observation operator $\mathbf{H}$.

Equation (30) serves as the reference algorithm hereafter, and we will compare its performance with that of the CHOP workflow in a number of different experiments below. In the comparison, we do not adopt any tailored methods proposed in the literature to tune δ and/or λ. Instead, we use the grid search method to find the optimal values of the pair $(\delta_{min}, \lambda_{min})$, whereas the optimality is meant in the sense that the combination $(\delta_{min}, \lambda_{min})$ results in the lowest value of an average RMSE within some pre-defined search ranges of δ and λ. In all the experiments related to the 40-dimensional L96 model, for the reference algorithm (Equation 30), the search range of δ is set to $\{0 : 0.1 : 2\}$, and that of λ to $\{0.05 : 0.05 : 1\}$. For a given experiment, the average RMSE is obtained by first computing the RMSEs of all analysis ensemble means at different time instances, then averaging these RMSEs over the whole assimilation time window, and finally averaging the previous (average) values again over a number of repetitions of the assimilation run. These repetitions share identical experimental settings, except that the random seeds used to generate certain random variables (e.g., the initial background ensemble and the observation noise) in each repetition of the experiment are different. In each experiment with respect to the 40-dimensional L96 model, the number of repetitions is set to 20.

In the CHOP workflow, instead of relying on the grid search method to find an optimal combination of δ and λ, the IES algorithm presented in Section 2 is applied to estimate an ensemble of δ and λ values for the reference algorithm (Equation 30). Note that there are differences between the optimality criterion used in the grid search method and that in the CHOP workflow. In this regard, the grid search method aims to find a single optimal pair $(\delta_{min}, \lambda_{min})$ that leads to the globally minimum average RMSE in the model space, within the whole assimilation time window. In contrast, the CHOP workflow searches for an ensemble of δ and λ values that help reduce the average of an ensemble of data mismatch values in the observation space (cf. Equation 3) within a given number of iteration steps, and at each data assimilation cycle (rather than the whole assimilation time window). In this sense, the obtained ensemble of δ and λ values represents, at best, locally optimal estimates at a given time instance, with a prescribed maximum number of iteration steps.

With these aforementioned differences, it is natural to expect that the globally optimal criterion (global criterion for short) used in the grid search method should result in better data assimilation performance than the locally optimal one (local criterion for short) adopted in the CHOP workflow. On the other hand, it is important to notice that the superiority of the global criterion is achieved on top of the assumption that one has access to the ground truths of model state variables and/or parameters during the whole data assimilation window. As such, it is not a realistic criterion that can be applied to practical data assimilation problems, where the underlying ground truths are typically unknown. In contrast, the local criterion is more realistic and can be implemented in practice. In the experiments below, however, we still choose to present the results with respect to the global criterion, as this serves as a means to cross-validate the performance of the CHOP workflow.

In the CHOP workflow, the configuration of the IES algorithm is as follows: Equations (15) and (16) are employed to estimate ensembles of hyper-parameters $\left\{\boldsymbol{\theta}_j^i \equiv \left[\delta_j^i, \lambda_j^i\right]^T\right\}_{j=1}^{N_e}$ at different iteration steps (indexed by i, for $i = 1, 2, \ldots, K$), and correlation-based localization is applied to Equation (15) (in addition to distance-based localization adopted in the reference algorithm, Equation 30). We note that the size of a hyper-parameter ensemble is the same as that of a background ensemble $\mathcal{M}^b = \{\mathbf{m}_j^b\}_{j=1}^{N_e}$ of model state variables and/or parameters, so that each ensemble member $\mathbf{m}_j^b$ is associated with its respective hyper-parameter pair $\left(\delta_j^i, \lambda_j^i\right)$, when using the reference algorithm (Equation 30) to update $\mathbf{m}_j^b$. To start the iteration process of the CHOP workflow, Latin hypercube sampling (LHS) is adopted to generate an initial ensemble of hyper-parameters at each assimilation cycle, whereas the hyper-parameter ranges used for LHS are the same as those in the grid search method.

Another remark is that the background ensemble $\mathcal{M}^b$ already exists before the CHOP workflow starts, and is invariant during the iteration process of the CHOP workflow. On the other hand, the outputs of the reference algorithm (Equation 30) do depend on the values of $\left(\delta_j^i, \lambda_j^i\right)$, and can change as the iteration proceeds. The members $\mathbf{m}_j^a$ of the analysis ensemble are taken as the outputs of Equation (30) at the last iteration step K, which is a number jointly determined by the three stopping criteria mentioned previously (cf. Section 2).

3.1.2. Results with different ensemble sizes

We first present results in a set of four experiments to illustrate the impacts of ensemble size. In each experiment, all state variables are observed (called full observation scenario hereafter), corresponding to the observation-index increment $\Delta n = 1$, with an observation frequency of every 4 integration steps (denoted by $N^{freq} = 4$). These four experiments use ensemble sizes $N_e = 15, 20, 25, 30$, respectively, while the remaining experimental settings (e.g., real observations/perturbed observations, initial background ensemble) are identical.

Figure 1 shows the average RMSEs in the full observation scenario, obtained by applying the grid search method to the reference algorithm (Equation 30), when different ensemble sizes N_e are used in the experiments.

For a given ensemble size, the sub-plots of Figure 1 indicate that in general, relatively low average RMSEs are reached with suitable amounts of covariance inflation and localization, whereas relatively high average RMSEs are obtained if there are insufficient inflation (corresponding to relatively small δ values) and localization (corresponding to relatively large λ values). On the other hand, too strong inflation (corresponding to relatively large δ values) and localization (corresponding to relatively small λ values) may lead to filter divergence (represented by white color in the sub-plots)[2], which corresponds to the situation where the RMSE values blow up with a potential issue of numerical overflow.

On the other hand, comparing the sub-plots of Figure 1, it can be observed that a larger ensemble size tends to result in a larger area that is filled with relatively low average RMSEs, while reducing the chance of filter divergence.

In company with Figure 1, Table 1 reports the minimum average RMSEs that the grid search method can achieve in the four sets of experiments, their associated STDs (to reflect the degrees of fluctuations of the average RMSEs within 20 repetition runs), and the optimal combinations $(\delta_{min}, \lambda_{min})$ of the inflation factor and the length scale, with which the minimum average RMSEs are achieved. As one can see therein, when the ensemble size increases, the minimum average RMSE obtained by the grid search method tends to decrease. Meanwhile, less amounts of covariance inflation (in the sense of smaller δ_{min}) and localization (in the sense of larger λ_{min}) are required to achieve the minimum average RMSE, consistent with the observations in Figure 1.

For comparison, Table 1 also lists the average RMSEs that are obtained by the CHOP workflow in the full observation scenario. Note that the CHOP workflow uses the IES to estimate an ensemble of inflation factors and length scales at each assimilation cycle. As such, unlike the grid search method, there

[2] If filter divergence takes place in any repetition run, then we assign NaN (not a number) to the average RMSE.

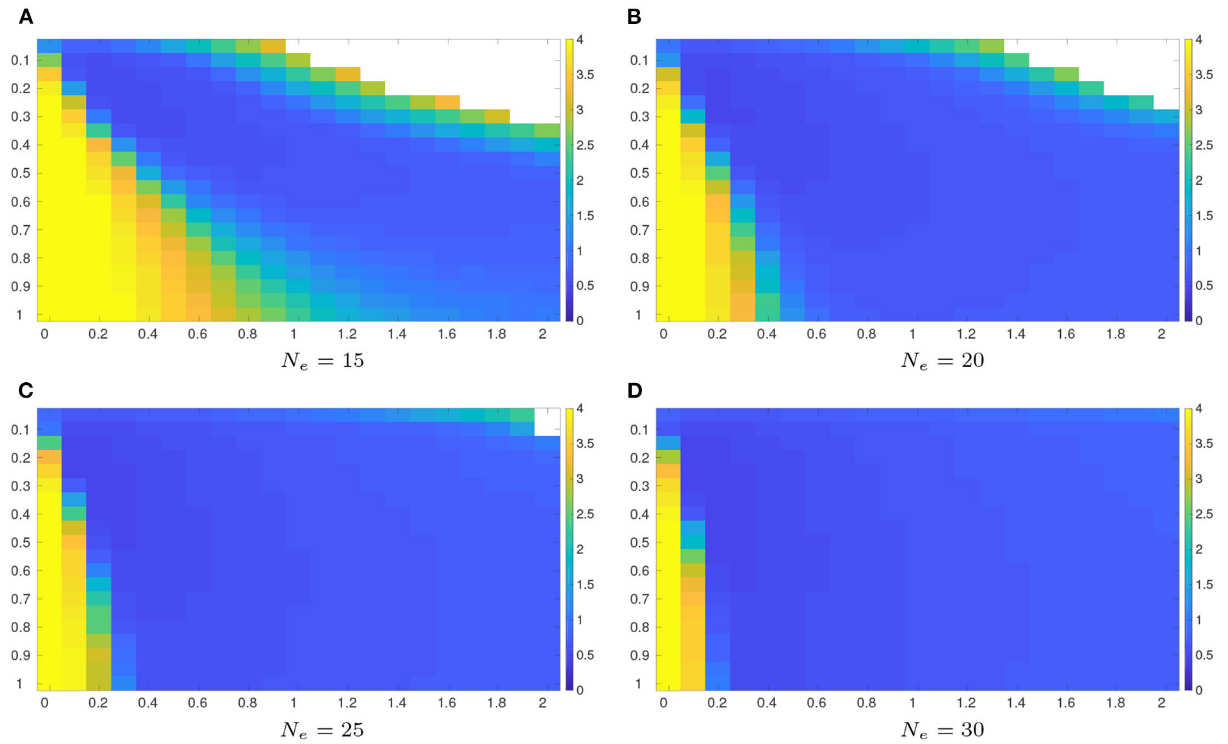


FIGURE 1

Average RMSEs with respect to the reference algorithm (Equation 30) in the full observation scenario ($\Delta n = 1$, $N^{freq} = 4$), using an ensemble size of 15, 20, 25, and 30, respectively. The RMSE values are obtained by searching all the possible combinations of the inflation factor $\delta \in \{0 : 0.1 : 2\}$ (along the horizontal axis) and the length scale $\lambda \in \{0.05 : 0.05 : 1\}$ (along the vertical axis). Note that for certain combinations of δ and λ values, filter divergence may take place (represented by white color in respective sub-plots). **(A)** N_e = 15. **(B)** N_e = 20. **(C)** N_e = 25. **(D)** N_e = 30.

TABLE 1 **Performance comparison between the grid search method and the CHOP workflow applied to the reference algorithm (Equation 30) in the full observation scenario, with four different ensemble sizes.**

Ensemble size	Grid search		CHOP
	Minimum average RMSE (mean ± STD)	$(\delta_{min}, \lambda_{min})$	Average RMSE (mean ± STD)
$N_e = 15$	0.5235 ± 0.0104	$(0.15, 0.15)$	1.2212 ± 0.1832
$N_e = 20$	0.4845 ± 0.0112	$(0.15, 0.25)$	0.6180 ± 0.0353
$N_e = 25$	0.4711 ± 0.0059	$(0.15, 0.30)$	0.5080 ± 0.0167
$N_e = 30$	0.4560 ± 0.0100	$(0.10, 0.20)$	0.4766 ± 0.0096

For the grid search method, we report the minimum average RMSEs within the search ranges, and their associated STDs. In addition, we also present the combination of the inflation factor and the length scale, $(\delta_{min}, \lambda_{min})$, that results in the minimum average RMSE in each experiment. For the CHOP workflow, the inflation factor and the length scale are estimated at each assimilation cycle, and thus vary with time. As such, we only report the average RMSEs and their associated STDs.

is no time-invariant, globally optimal inflation factor or length scale obtained from the CHOP workflow.

A few observations can be obtained when comparing the performance of the grid search method and the CHOP workflow in Table 1. First of all, in terms of the minimum average RMSE that one can achieve in each experiment, the CHOP workflow systematically under-performs the grid search method. This under-performance is not surprising, since, as discussed previously, the grid search method gains the relative superiority on top of the assumption that it has access to the ground truths, which is typically infeasible in practical data assimilation problems.

In comparison to the grid search method, the CHOP workflow appears to be more sensitive to the change of ensemble size. With $N_e = 15$, there is a relatively large gap (around 0.7) between the average RMSE of the CHOP workflow and the minimum average RMSE that the grid search method can achieve. As the ensemble size increases, the performance of the CHOP workflow substantially improves, such that the gap drops to only around 0.02 when $N_e = 30$. This indicates that in the full observation scenario, the CHOP workflow can perform reasonably well with a sufficiently large ensemble size.

A number of factors, including model, data, and ensemble sizes, observation frequency and density, the number of hyper-parameters and the searching ranges of their values, would have an influence on the computational time required to deploy the grid search method and/or the CHOP workflow. As such, instead of presenting the computational time in all possible combinations of these different factors, we compare the computational time with respect to a normal EnKF equipped with a specific combination of the inflation factor $\delta = 0$ and the length scale $\lambda = 0.1$ (corresponding to a single grid point in the grid search method), and that with respect to the EnKF equipped with the CHOP workflow. In this comparison, the ensemble size $N_e = 30$ ($\Delta n = 1$ and $N^{freq} = 4$), and our computing system uses Intel(R) Core(TM) i9-10900K CPU @ 3.70 GHz with 64 GB memory. Under these settings, the wall-clock time for the normal EnKF is 15.9261 ± 0.4102 (mean ± STD) seconds, while the wall-clock time for the EnKF equipped with the CHOP workflow is 65.5915 ± 0.6852 s.

As mentioned in Section 2.2, for the EnKF with the CHOP workflow, the maximum number of iteration steps in the IES algorithm is set to 10, which means that the maximum computational time at the analysis step of the EnKF equipped with the CHOP workflow is around 10 times that at the analysis step of the normal EnKF. From the above-reported results, however, it appears that on average the computational time of the EnKF equipped with the CHOP workflow is around 4.1 times that of the normal EnKF, which is substantially lower than 10. This difference may be attributed to the following two factors: (1) The IES may stop before reaching the maximum number of iteration steps, due to the other two stopping criteria specified in Section 2.2; (2) The reported computational cost includes the time at both the forecast and the analysis steps of the EnKF during the whole assimilation time window. While the EnKF with the CHOP workflow has a higher computational cost at an analysis step than the normal EnKF, at a forecast step they would have roughly the same computational cost instead.

Note that so far we have only compared the computational time between the normal EnKF (at a single grid point) and the EnKF equipped with the CHOP workflow. When the grid search method is applied to find the optimal combination of hyper-parameters, the total computational cost is roughly equal to the number of grid points times the cost of a single normal EnKF. In the current experiment setting, the grid search method considers 21 values for the inflation factor, and 20 values for the length scale. As such, it needs to compare the results at 21 × 20 grid points (hence 420 normal EnKF runs) in one repetition of the experiment. Therefore, under this setting, the grid search method will be roughly 100 times more expensive than the EnKF with the CHOP workflow. It is expected that similar conclusions would be obtained under other experiment settings, but for brevity we do not present further comparison results in this regard.

3.1.3. Results with different observation densities

We then examine the impact of observation density on the performance of the grid search method and the CHOP workflow. To this end, we conduct three more experiments with the observation-index increment $\Delta n = 2$ (the half observation scenario), $\Delta n = 4$ (the quarter observation scenario), $\Delta n = 8$ (the octantal observation scenario), respectively, while these three experiments share the same ensemble size $N_e = 30$ and observation frequency $N^{freq} = 4$.

Figure 2 reports the average RMSEs with different combinations of the inflation factor and length scale values, obtained by the grid search method in the half, quarter and octantal observation scenarios, respectively. For convenience of comparison, the results of the full observation scenario (with $N_e = 30$) in Figure 1D are re-plotted therein. Comparing the results in Figure 2, it can be seen that, as the observation density decreases (Δn increases), the performance of the grid search method degrades, in the sense that the resulted average RMSEs arise, and filter divergence tends to have a higher chance to take place, except that the quarter observation scenario seems to have more instances of filter divergence than the octantal observation scenario. The degraded performance is expected, since reduced observation density means that less information contents can be utilized for data assimilation.

Similar to Tables 1, 2 posts the minimum average RMSEs of the grid search method, their associated STDs, and the optimal values of the inflation factor and the length scale. Among the full, half and quarter observation scenarios, as the observation density decreases, the optimal inflation factor δ_{min} does not change, but the optimal length scale λ_{min} shows a tendency of increment, meaning that less localization is required. This trend, however, is broken in the octantal observation scenario, in which both δ_{min} and λ_{min} become smaller than those of the other three scenarios, suggesting that it is better to have less inflation but more localization.

For comparison, Table 2 also lists the average RMSEs with respect to the CHOP workflow. As one can see therein, in different observation scenarios, the average RMSEs of the CHOP workflow stay in a relatively close vicinity of the minimum values achieved by the grid search method. In addition, no filter divergence is spotted in the repetition runs of the CHOP workflow. As such, the CHOP workflow again appears to work reasonably well with different observation densities.

3.1.4. Results with different observation frequencies

We investigate one more aspect, namely, the impact of observation frequency on the performance of the grid search method and the CHOP workflow. In line with this goal, we conduct three additional experiments, with the following

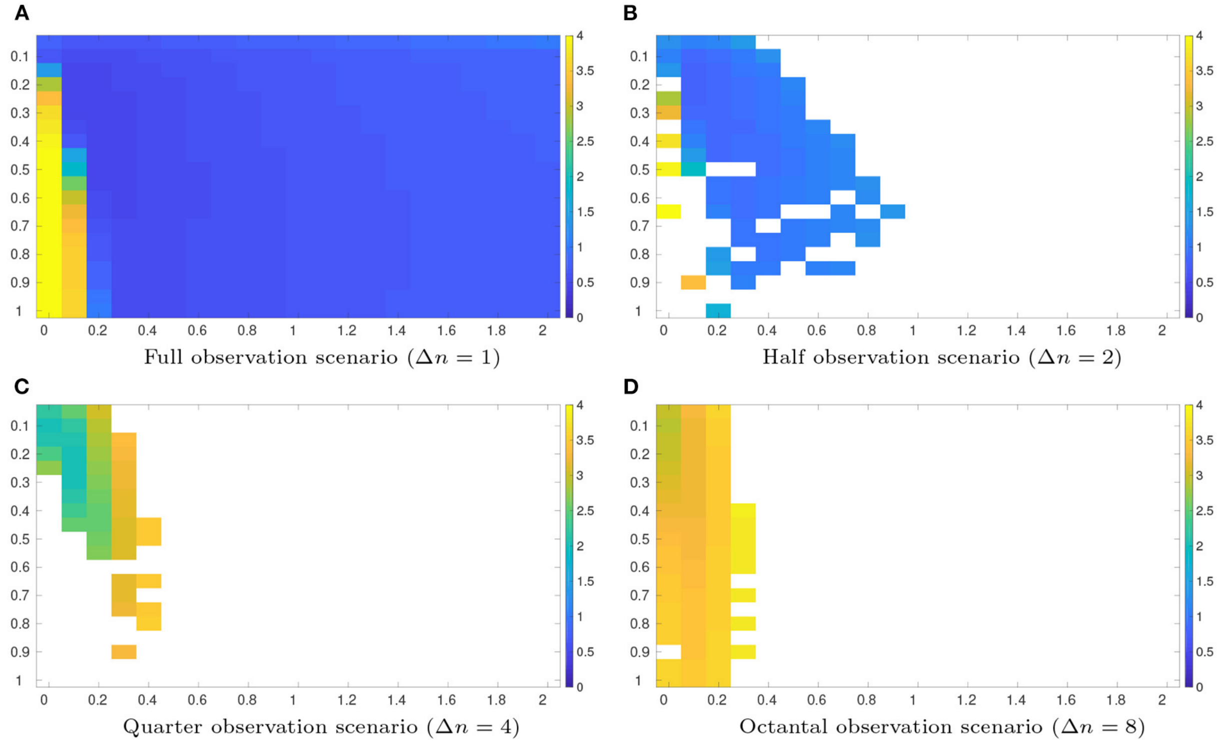


FIGURE 2
As in Figure 1, but for average RMSEs obtained by the grid search method in the half ($\Delta n = 2$, $N^{freq} = 4$), quarter ($\Delta n = 4$, $N^{freq} = 4$), and octantal ($\Delta n = 8$, $N^{freq} = 4$) observation scenarios, respectively, with the ensemble sizes $N_e = 30$. For ease of comparison, the results of the full observation scenario ($\Delta n = 1$, $N^{freq} = 4$, $N_e = 30$) in Figure 1 are re-plotted here. **(A)** Full observation scenario (Δn = 1). **(B)** Half observation scenario (Δn = 2). **(C)** Quarter observation scenario (Δn = 4). **(D)** Octantal observation scenario (Δn = 8).

TABLE 2 As in Table 1, but for performance comparison between the grid search method and the CHOP workflow with full, half, quarter, and octantal observations, respectively, whereas the ensemble size and the observation frequency are set to 30 and 4, respectively, in all experiments.

Observation density	Grid search		CHOP
	Minimum average RMSE (mean ± STD)	$(\delta_{min}, \lambda_{min})$	Average RMSE (mean ± STD)
Full ($\Delta n = 1$)	0.4560 ± 0.0100	$(0.10, 0.20)$	0.4766 ± 0.0096
Half ($\Delta n = 2$)	0.7975 ± 0.0257	$(0.10, 0.20)$	0.8763 ± 0.0418
Quarter ($\Delta n = 4$)	2.0100 ± 0.0773	$(0.10, 0.25)$	2.3596 ± 0.1248
Octantal ($\Delta n = 8$)	2.9129 ± 0.0353	$(0.05, 0.10)$	3.2437 ± 0.0419

settings: $N_e = 30$, $\Delta n = 2$ (the half observation scenario), and $N^{freq} = 1, 2, 8$, respectively.

Figure 3 shows the average RMSEs of the grid search method, when the inflation factor and the length scale take different values, and the observations arrive at different frequencies. For convenience of comparison, the results with $N^{freq} = 4$ ($N_e = 30$, $\Delta n = 2$) in Figure 2 are also included into Figure 3. It can be clearly seen that, as the observation frequency decreases (corresponding to increasing N^{freq}), the average RMSE tends to increase. Filter divergence remains a problem, but in this case, it appears that a lower observation frequency does not necessarily lead to a higher chance of filter divergence.

Following Tables 1–3 summarizes the minimum average RMSEs of the grid search method at different observation frequencies, their associated STDs and the optimal inflation factor and length scale. As observed in Table 3, when the observation frequency decreases (N^{freq} increases), the minimum average RMSE arises. In the meantime, the corresponding optimal length scale λ_{min} tends to decline, while the optimal inflation factor δ_{min} remains unchanged.

TABLE 3 As in Table 1, but for performance comparison between the grid search method and the CHOP workflow in the half observation scenario ($\Delta n = 2$), with the same ensemble size $N_e = 30$ yet different observation frequencies.

Observation frequency	Grid search		CHOP
	Minimum average RMSE (mean ± STD)	$(\delta_{min}, \lambda_{min})$	Average RMSE (mean ± STD)
$N^{freq} = 1$	0.3948 ± 0.0124	$(0.10, 0.45)$	0.5409 ± 0.0117
$N^{freq} = 2$	0.5015 ± 0.0123	$(0.10, 0.30)$	0.5471 ± 0.0193
$N^{freq} = 4$	0.7975 ± 0.0257	$(0.10, 0.20)$	0.8763 ± 0.0418
$N^{freq} = 8$	1.8369 ± 0.0557	$(0.10, 0.20)$	2.1022 ± 0.0473

In terms of the performance of the CHOP workflow, one can observe again that its average RMSEs stay relatively close to the corresponding minimum values of the grid search method. On the other hand, no filter divergence is found in the repetition runs of the CHOP workflow. Altogether, the experiment results confirm that the CHOP workflow also performs reasonably well at different observation frequencies.

3.2. Experiments in a 1,000-dimensional L96 system

In this subsection, we conduct an additional experiment in a 1,000-dimensional L96 model (N_L =1,000). The main purpose of the experiment is to demonstrate that the CHOP workflow can be used to tune a large number of hyper-parameters. This feature is a natural reflection of the capacity of the IES algorithm, which has been shown to work well in, e.g., large-scale reservoir data assimilation problems [20–22].

The experiment settings in this subsection is largely the same as those of the experiments with respect to the 40-dimensional L96 model. Therefore, for brevity, in the sequel we focus more on explaining the places where different experiment settings are adopted.

Since the dimensionality is significantly increased, the grid search method becomes more time-consuming. To facilitate the investigation, we reduce the assimilation time window from 250 time units to 100 time units (corresponding to 2,000 integration steps), and the number of repetition runs of a given experiment from 20 to 10, while keeping the search ranges of the inflation factor and the length scale unchanged. In the meantime, we increase the ensemble size N_e to 100. The observation system is the same as that in Equation (26), with the same observation-noise variance. The increment of model-variable index is set to $\Delta n = 4$ (quarter observation scenario), and the observations are collected every four integration steps ($N^{freq} = 4$). Given the purpose of the current experiment, no sensitivity study (e.g., with respect to N_e, Δn and N^{freq}) is conducted.

The base assimilation algorithm is the same as that in Equation (27), and we introduce both covariance inflation and localization to the base algorithm. We use the same localization scheme as in the 40-dimensional case (with the length scale λ as a hyper-parameter), while considering two different ways of conducting covariance inflation. One inflation method is again the same as that in the 40-dimensional case, which applies a single inflation factor δ to all model state variables of the background ensemble. This leads to a reference algorithm identical to that in Equation (30), which contains two hyper-parameters, δ and λ, and the grid search method is then applied to find the optimal combination of δ and λ for the reference algorithm. On the other hand, the CHOP workflow is employed to estimate an ensemble of N_e hyper-parameter pairs $\left\{(\delta_j, \lambda_j)\right\}_{j=1}^{N_e}$. For distinction later, we call the application of the CHOP workflow to estimate the ensemble $\left\{(\delta_j, \lambda_j)\right\}_{j=1}^{N_e}$ the single-inflation-factor (SIF) method.

The other inflation method introduces multiple inflation factors to the base algorithm. Specifically, each model state variable of the background ensemble $\mathcal{M}^b = \{\mathbf{m}_j^b\}_{j=1}^{N_e}$ receives its own inflation factor, in such a way that after inflation, the modified background ensemble $\tilde{\mathcal{M}}^b \equiv \{\tilde{\mathbf{m}}_j^b\}_{j=1}^{N_e}$ has its member $\tilde{\mathbf{m}}_j^b$ in the form of $\tilde{\mathbf{m}}_j^b = \bar{\mathbf{m}}^b + (\mathbf{1} + \boldsymbol{\delta}) \circ \left(\mathbf{m}_j^b - \bar{\mathbf{m}}^b\right)$, where $\mathbf{1}$ is a N_L-dimensional vector with all its elements equal to 1, $\boldsymbol{\delta} = \left[\delta_1, \delta_2, \cdots, \delta_{N_L}\right]^T$ contains N_L inflation factors, and $\circ$ stands for the Schur product operator. Replacing the SIF method in Equation (30) by the multiple-factor one (while keeping the localization scheme unchanged), one obtains a new reference algorithm.

$$\mathbf{m}_j^a = \tilde{\mathbf{m}}_j^b + \left\{\mathbf{L}(\lambda) \circ \left[\tilde{\mathbf{C}}_m \mathbf{H}^T \left(\mathbf{H}\tilde{\mathbf{C}}_m \mathbf{H}^T + \mathbf{C}_d\right)^{-1}\right]\right\} \left(\mathbf{d}_j^o - \mathbf{H}\tilde{\mathbf{m}}_j^b\right); \tag{31}$$

$$\tilde{\mathbf{m}}_j^b = \bar{\mathbf{m}}^b + (\mathbf{1} + \boldsymbol{\delta}) \circ \left(\mathbf{m}_j^b - \bar{\mathbf{m}}^b\right), \tag{32}$$

where $\tilde{\mathbf{C}}_m$ is the sample covariance matrix with respected to the inflated ensemble $\tilde{\mathcal{M}}^b$.

Due to the high dimensionality (N_L =1000), it is computationally prohibitive to apply the grid search method to optimize the set of hyper-parameters in Equation (31). On

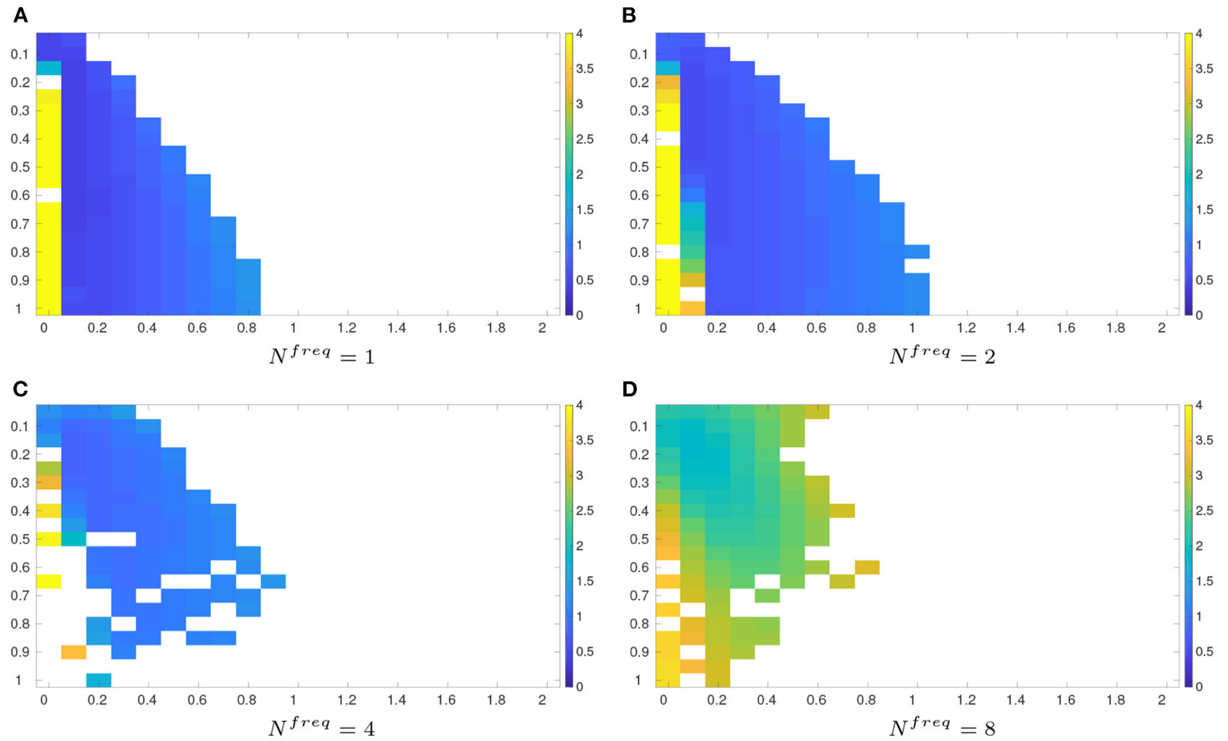


FIGURE 3
As in Figure 1, but for average RMSEs obtained by the grid search method in the half observation scenario, with the same ensemble sizes $N_e = 30$ yet different observation frequencies. For ease of comparison, the results of the half observation scenario ($\Delta n = 2$, $N^{freq} = 4$, $N_e = 30$) in Figure 2 are re-plotted here. **(A)** $N^{freq} = 1$. **(B)** $N^{freq} = 2$. **(C)** $N^{freq} = 4$. **(D)** $N^{freq} = 8$.

the other hand, as will be shown later, it is still possible to apply the CHOP workflow to estimate an ensemble of hyper-parameters, denoted by $\left\{(\delta_j, \lambda_j)\right\}_{j=1}^{N_e}$. Such a workflow is called the multiple-inflation-factor (MIF) method hereafter.

With these said, in the sequel, we compare the performance of the grid search method applied to the reference algorithm in Equation (30), the CHOP workflow with the SIF method, and the CHOP workflow with the MIF method, respectively.

Figure 4 shows the average RMSEs obtained by the grid search method with different combinations of δ and λ values. Similar to what we have seen in the 40-dimensional L96 model, filter divergence arises in a large portion of the searched region of hyper-parameters. As reported in Table 4, the minimum average RMSE of the grid search method is around 2.7667, achieved at $\delta_{min} = 0.10$ and $\lambda_{min} = 0.05$.

For comparison, Table 4 also presents the average RMSEs of the CHOP workflow equipped with the SIF and MIF methods, respectively. Again, no filter divergence takes place in the CHOP workflow. Both the SIF and MIF methods result in RMSE values that stay relatively close to the minimum RMSE value of the grid search method. In comparison to the SIF method, however, the MIF exhibits better performance, largely due to a higher degree of freedom brought in by the larger number of inflation factors used in the assimilation algorithm.

3.3. Behavior of the IES algorithm

Finally we take a glance at the behavior of the IES algorithm that underpins the CHOP workflow. We do this in the 1,000-dimensional L96 model with the MIF method, to illustrate the efficacy of the IES algorithm in dealing with high-dimensional problems. Note that in the CHOP workflow, the IES is adopted to tune hyper-parameters at each assimilation cycle. For brevity, we only use one of the assimilation cycles for illustration.

Figures 5, 6 disclose the data mismatch and RMSE values at each iteration step, in the form of box plots. These values are obtained as follows: At each iteration step, we first insert the ensemble of hyper-parameters into the reference algorithm (Equation 31) of the MIF method, in such a way that each member of the background ensemble (of model state variables) is associated with a member of the ensemble of hyper-parameters. In this way, we obtain an ensemble

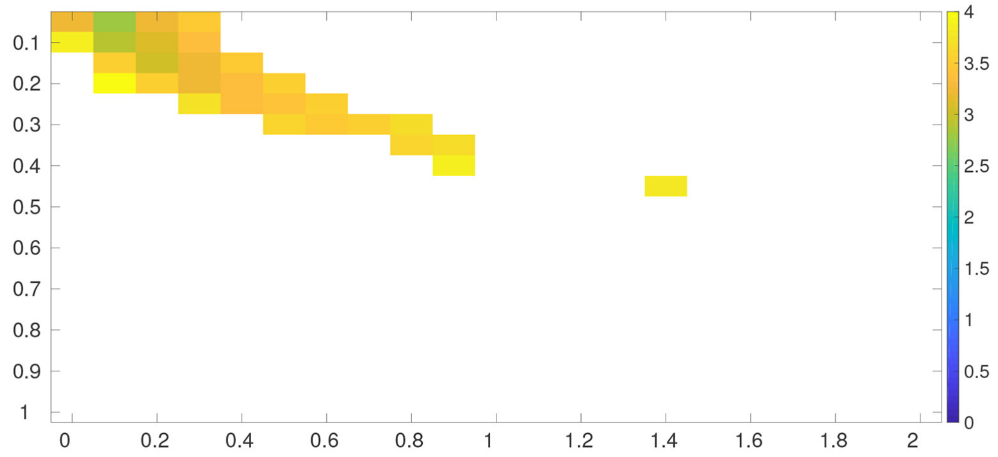


FIGURE 4
Average RMSEs obtained by the grid search method (applied to Equation 30) in the 1,000-dimensional L96 model.

TABLE 4 Performance comparison between the grid search method and the CHOP workflow in the 1,000-dimensional L96 model.

Grid search		CHOP (SIF)	CHOP (MIF)
Minimum average RMSE (mean ± STD)	**$(\delta_{min}, \lambda_{min})$**	**Average RMSE (mean ± STD)**	**Average RMSE (mean ± STD)**
2.7667 ± 0.0099	(0.10, 0.05)	3.4213 ± 0.0552	3.0264 ± 0.0116

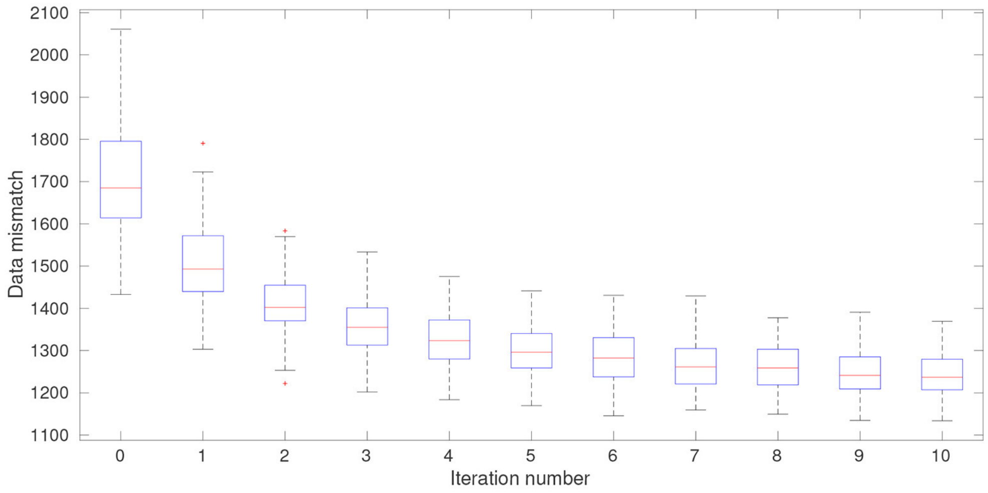


FIGURE 5
Box plots of data mismatch at different iteration steps at one of the data assimilation cycles of the 1,000-dimensional L96 model.

of updated model state variables at each iteration step. The data mismatch and RMSE values are then calculated with respect to the ensemble of updated model state variables. Note that the ensemble of analysis state variables corresponds to the ensemble of updated model state variables at the last iteration step. Meanwhile, at iteration step 0, the data mismatch and RMSE values are computed based on the initial ensemble of hyper-parameters generated through the LHS scheme.

In Figures 5, 6, both the data mismatch and RMSE values tend to decrease as the iteration proceeds, while maintaining substantial ensemble varieties in the box plots (indicating that ensemble collapse does not take place). The IES converges relatively fast, moving into the

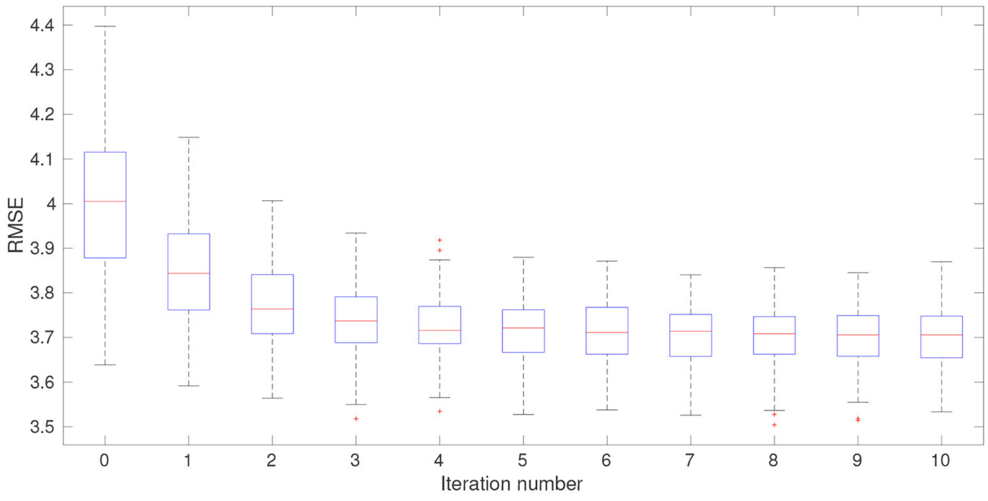


FIGURE 6
Box plots of RMSE at different iteration steps at one of the data assimilation cycles of the 1,000-dimensional L96 model.

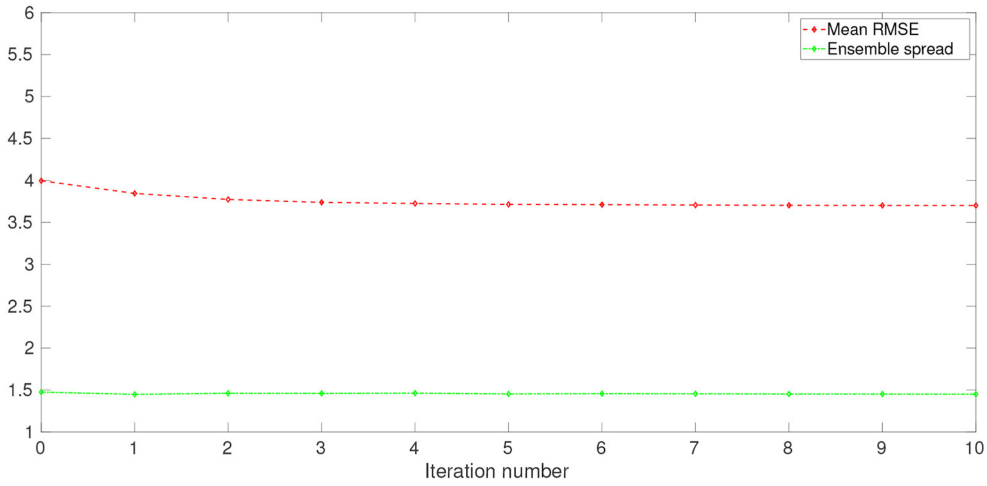


FIGURE 7
Mean RMSE (dashed red line) and ensemble spread (dash-dotted green line) vs. iteration step, at one of the data assimilation cycles of the 1,000-dimensional L96 model.

vicinity of a certain local minimum after only several iteration steps, which is a behavior also noticed in other studies [20–22].

Corresponding to Figures 5–7 presents the values of mean RMSE and ensemble spread at each iteration step. Here, a mean RMSE is the average of the RMSEs over ensemble members of the updated model state variables (i.e., the average of the box-plot values) at a given iteration step, whereas ensemble spread is evaluated according to Equation (25). In consistency with Figure 6, the mean RMSE and the ensemble spread tend to decrease along with the iterations. The overall change of ensemble spread from the beginning to the end of the iteration process appears to be less significant than that of the mean RMSE. In fact, the final ensemble spread appears to stay close to the initial value, which also suggests that ensemble collapse does not appear to be a problem. On the other hand, there are substantial gaps between the values of mean RMSE and ensemble spread at all iteration steps, which means that ensemble spread does not match the estimation errors of the updated model state variables. This tendency of under-estimation seems to be largely related to the fact that the ensemble spread at the beginning of the iteration is already considerably smaller than the mean RMSE, which could be due to the insufficient ensemble spread in

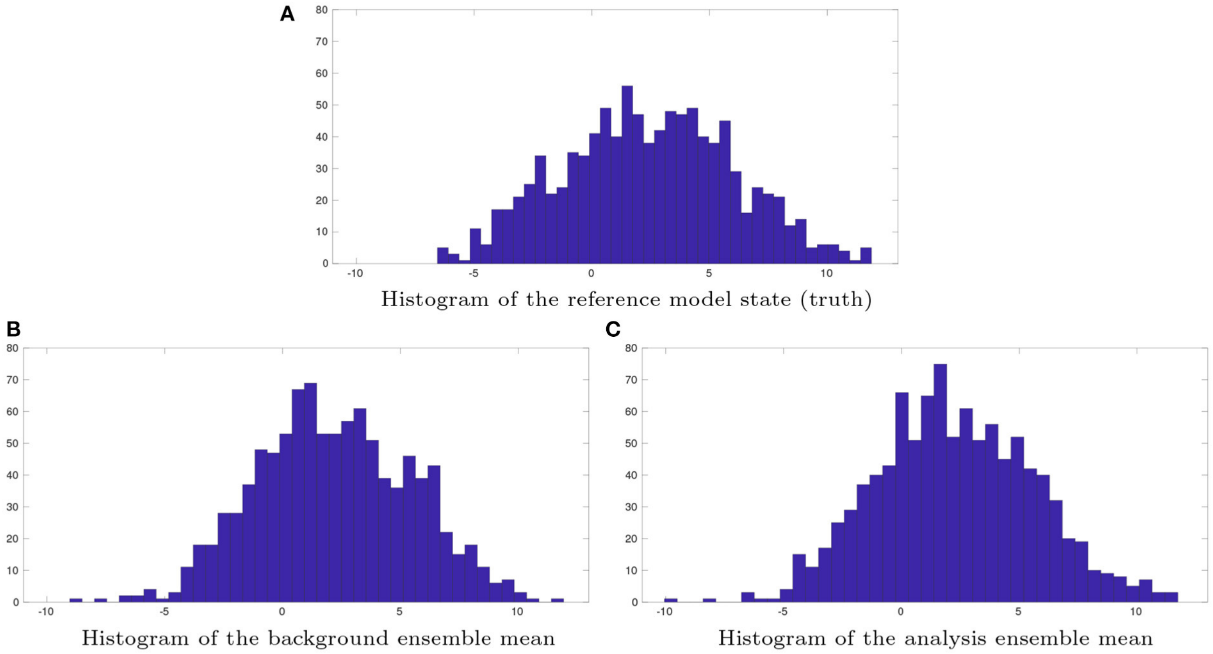


FIGURE 8
Histograms of **(A)** the reference model state (truth), **(B)** the background ensemble mean, and **(C)** the analysis ensemble mean at one of the assimilation cycles in the 1,000-dimensional L96 model. Both the reference model state and the background ensemble do not change over the IES iteration process, whereas the analysis ensemble is obtained by inserting the ensemble of estimated hyper-parameters at the last iteration step into the reference algorithm (Equation 31), of the MIF method.

the background ensemble, or the initial ensemble of hyper-parameters, or both.

Figure 8 shows the histograms with respect to the reference model state variables (the truth), the background-ensemble mean, and the analysis-ensemble mean, respectively. It is clear that neither the histogram of the background-ensemble mean, nor that of the analysis-ensemble mean, resemble the histogram of the truth well, suggesting that there are substantial estimation errors in the estimated model state variables.

On the other hand, the results with respect to the estimated hyper-parameters appears to be more interesting. For illustration, Figure 9 plots the histograms of the initial (left) and final (right) ensembles of the inflation factors associated with model state variable 1 (top) and 500 (middle), and the histograms of the initial and final ensembles of the length scale (bottom). Since we use LHS to generate the initial ensemble, it can be observed that the histograms with respect to three initial ensembles of hyper-parameters roughly follow certain uniform distributions. Through the iteration process of the IES algorithm, the shapes and supports of the histograms are modified. This is particularly noticeable for the estimated values of length scale in the final ensemble (Figure 9F). Initially, the range of the length scale in the initial ensemble is $[0.05, 1]$, at the end of the iteration, around 80% of the values of estimated length scale locate at 0.05 (which is the optimal value found by the grid search method), while the rest of the estimated values are less than 0.1. On the other hand, for the estimated inflation factors, one may notice that their values are less concentrated than the length scale. In comparison to the initial ensembles of the inflation factors, their final ensembles receive somewhat narrower supports, but still maintain sufficient spreads, in consistency with the results in Figure 7. The values of estimated inflation factors are substantially larger than the optimal inflation factor (0.10) found by the grid search method. The main reason behind this is that the original EnKF updates model state variables only once, whereas the CHOP workflow does the update multiple times, each time with a smaller step size (hence larger inflation factors).

4. Discussion and conclusion

This study aims to develop a Continuous Hyper-parameter OPtimization (CHOP) workflow that helps to tune hyper-parameters in ensemble data assimilation algorithms. The main idea is to treat a data assimilation algorithm with certain hyper-parameters as a parametric mapping that transforms an ensemble of initial model state variables and/or parameters to a corresponding ensemble of updated quantities, which in

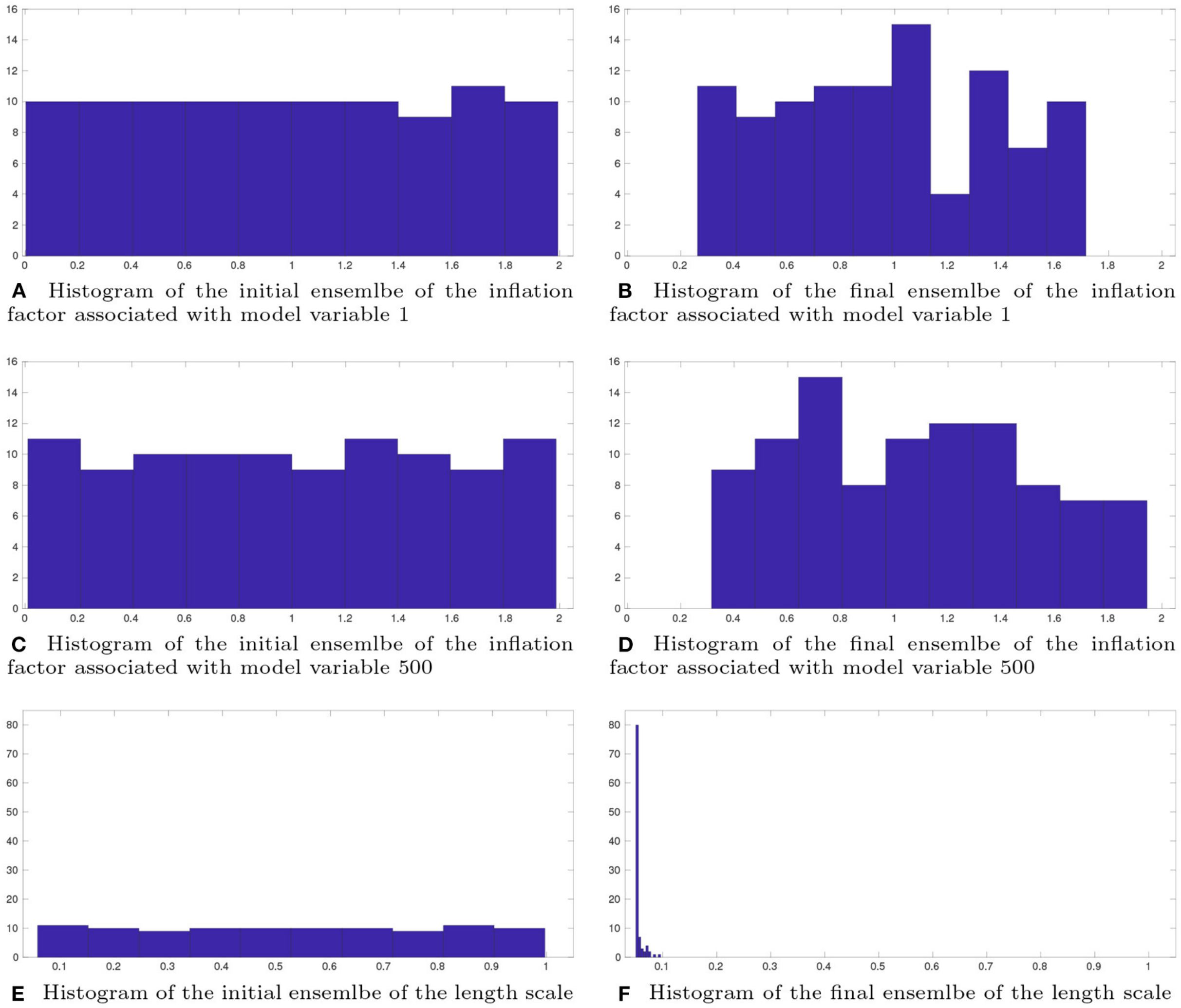


FIGURE 9
Histograms of the initial (left) and final (right) ensembles, with respect to the inflation factors associated with model state variables 1 **(A,B)** and 500 **(C,D)**, and the length scale **(E,F)**, respectively.

turn are related to the predicted observations through the observation operator.

Following this perspective, the hyper-parameters can be tuned in such a way that the corresponding updated model state variables and/or parameters result in lower data mismatch than their initial values. In doing so, the CHOP problem is recast as a parameter estimation problem. We adopt an iterative ensemble smoother (IES) to solve the CHOP problem, as its derive-free nature allows one to implement the algorithm without explicitly knowing the relevant gradients. To mitigate the adverse effects of using a relatively small ensemble size in the IES, we also equip the IES with a correlation-based adaptive localization scheme, which helps to handle the issue that hyper-parameters may not possess physical locations needed for distance-based localization schemes.

We investigate the performance of the CHOP workflow in the Lorentz 96 (L96) model with two different dimensions. Experiments in the 40-dimensional L96 model aim to inspect the impacts of a few factors on the performance of the CHOP workflow, whereas those in the 1,000-dimensional L96 model focus on demonstrating the capacity of the CHOP workflow to deal with a high-dimensional set of hyper-parameters, which may not be computationally feasible for the grid search method. Such a capacity would help enable the developments of more sophisticated auxiliary techniques (e.g., inflation or localization) that introduce a large number of hyper-parameters to an assimilation algorithm for further performance improvements.

In most of the experiments, the CHOP workflow is able to achieve reasonably good performance, which is relatively close to the best performance obtained by the grid search method (an unverifiable case occurs in the experiments with respect to the multiple-inflation-factor method in the 1,000-dimensional L96 model, where we are not able to adopt the grid search method due to its prohibitively expensive cost). Meanwhile, unlike the grid search method, the optimality criterion in the CHOP workflow is based on data mismatch between real and predicted observations, which is realistic and can be implemented in practical data assimilation problems.

So far, we have only implemented the CHOP workflow in the ensemble Kalman filter (EnKF) with perturbed observations. Given the varieties of different assimilation algorithms (some of them may not even be ensemble-based), the way of implementing a CHOP workflow may have to adapt to the particular assimilation algorithm in choice, which is an issue to be further studied in the future. On the other hand, though, we expect that the notion of treating an assimilation algorithm with hyper-parameters as a parametric mapping may still be valid. As such, it appears sensible that one converts a generic assimilation problem (being state estimation, parameter estimation or both) with hyper-parameters into a parameter estimation problem, and solve it through a certain iterative assimilation algorithm.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

XL: conception, experimentation, validation, and writing. C-AX: conception and validation. All authors contributed to the article and approved the submitted version.

Funding

XL acknowledges financial supports from the NORCE research project Assimilating 4D Seismic Data: Big Data Into Big Models which is funded by industry partners, Equinor Energy AS, Lundin Energy Norway AS, Repsol Norge AS, Shell Global Solutions International B.V., TotalEnergies EP Norge AS, and Wintershall Dea Norge AS, as well as the Research Council of Norway (project number: 295002). C-AX acknowledges financial supports from the National Nature Science Foundation of China (Grant No. 42002247) and the Nature Science Foundation of Guangdong Province, China (Grant No. 2020A1515111054).

Acknowledgments

The authors would like to thank reviewers for their constructive and insightful suggestions and comments, which help to significantly improve the quality of the current work.

Declarations

The code for the iterative ensemble smoother (IES) and the Lorentz 96 model is available from https://github.com/lanhill/Iterative-Ensemble-Smoother. The ensemble Kalman filter (EnKF) is well-studied in the literature, and it is publicly available from a number of different sources, see, for example, https://se.mathworks.com/matlabcentral/fileexchange/31093-ensemble-kalman-filter for MATLAB code.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Kalman R. A new approach to linear filtering and prediction problems. *Trans ASME Ser D J Basic Eng.* (1960) 82:35–45. doi: 10.1115/1.3662552

2. Simon D. *Optimal State Estimation: Kalman, H-Infinity, and Nonlinear Approaches.* Hoboken, NJ: Wiley-Interscience (2006). doi: 10.1002/0470045345

3. Julier SJ, Uhlmann JK, Durrant-Whyte HF. A new approach for filtering nonlinear systems. In: *The Proceedings of the American Control Conference.* Seattle, WA (1995). p. 1628–32.

4. Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear and non-Gaussian Bayesian state estimation. *IEEE Proc Radar Signal Process.* (1993) 140:107–13. doi: 10.1049/ip-f-2.1993.0015

5. Van Leeuwen PJ. Particle filtering in geophysical systems. *Mon Weath Rev.* (2009) 137:4089–114. doi: 10.1175/2009MWR2835.1

6. Sorenson HW, Alspach DL. Recursive Bayesian estimation using Gaussian sums. *Automatica.* (1971) 7:465–79. doi: 10.1016/0005-1098(71)90097-5

7. Courtier P, Andersson E, Heckley W, Vasiljevic D, Hamrud M, Hollingsworth A, et al. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Q J R Meteorol Soc.* (1998) 124:1783–807. doi: 10.1002/qj.49712455002

8. Courtier P, Thépaut JN, Hollingsworth A. A strategy for operational implementation of 4D-Var, using an incremental approach.

Q J R Meteorol Soc. (1994) 120:1367–87. doi: 10.1002/qj.49712051912

9. Cohn SE, Sivakumaran N, Todling R. A fixed-lag Kalman smoother for retrospective data assimilation. *Mon Weath Rev.* (1994) 122:2838–67. doi: 10.1175/1520-0493(1994)122<2838:AFLKSF>2.0.CO;2

10. Evensen G. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J Geophys Res.* (1994) 99:10143–62. doi: 10.1029/94JC00572

11. Anderson JL. An ensemble adjustment Kalman filter for data assimilation. *Mon Weath Rev.* (2001) 129:2884–903. doi: 10.1175/1520-0493(2001)129<2884:AEAKFF>2.0.CO;2

12. Bishop CH, Etherton BJ, Majumdar SJ. Adaptive sampling with ensemble transform Kalman filter. Part I: theoretical aspects. *Mon Weath Rev.* (2001) 129:420–36. doi: 10.1175/1520-0493(2001)129<0420:ASWTET>2.0.CO;2

13. Hamill TM, Whitaker JS, Snyder C. Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. *Mon Weath Rev.* (2001) 129:2776–90. doi: 10.1175/1520-0493(2001)129<2776:DDFOBE>2.0.CO;2

14. Pham DT. Stochastic methods for sequential data assimilation in strongly nonlinear systems. *Mon Weath Rev.* (2001) 129:1194–207. doi: 10.1175/1520-0493(2001)129<1194:SMFSDA>2.0.CO;2

15. Hunt BR, Kostelich EJ, Szunyogh I. Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter. *Phys D.* (2007) 230:112–26. doi: 10.1016/j.physd.2006.11.008

16. Sakov P, Oliver DS, Bertino L. An iterative EnKF for strongly nonlinear systems. *Mon Weath Rev.* (2012) 140:1988–2004. doi: 10.1175/MWR-D-11-00176.1

17. Evensen G, van Leeuwen PJ. An ensemble Kalman smoother for nonlinear dynamics. *Mon Weath Rev.* (2000) 128:1852–67. doi: 10.1175/1520-0493(2000)128<1852:AEKSFN>2.0.CO;2

18. Van Leeuwen PJ, Evensen G. Data assimilation and inverse methods in terms of a probabilistic formulation. *Mon Weath Rev.* (1996) 124:2898–913. doi: 10.1175/1520-0493(1996)124<2898:DAAIMI>2.0.CO;2

19. Bocquet M, Sakov P. An iterative ensemble Kalman smoother. *Q J R Meteorol Soc.* (2014) 140:1521–35. doi: 10.1002/qj.2236

20. Emerick AA, Reynolds AC. Ensemble smoother with multiple data assimilation. *Comput Geosci.* (2012) 55:3–15. doi: 10.1016/j.cageo.2012.03.011

21. Chen Y, Oliver D. Levenberg-Marquardt forms of the iterative ensemble smoother for efficient history matching and uncertainty quantification. *Comput Geosci.* (2013) 17:689–703. doi: 10.1007/s10596-013-9351-5

22. Luo X, Stordal A, Lorentzen R, Nævdal G. Iterative ensemble smoother as an approximate solution to a regularized minimum-average-cost problem: theory and applications. *SPE J.* (2015) 20:962–82. doi: 10.2118/176023-PA

23. Ambadan JT, Tang Y. Sigma-point Kalman filter data assimilation methods for strongly nonlinear systems. *J Atmos Sci.* (2009) 66:261–85. doi: 10.1175/2008JAS2681.1

24. Luo X, Moroz IM. Ensemble Kalman filter with the unscented transform. *Phys D.* (2009) 238:549–62. doi: 10.1016/j.physd.2008.12.003

25. Hoteit I, Pham DT, Triantafyllou G, Korres G. A new approximate solution of the optimal nonlinear filter for data assimilation in meteorology and oceanography. *Mon Weath Rev.* (2008) 136:317–34. doi: 10.1175/2007MWR1927.1

26. Hoteit I, Luo X, Pham DT. Particle Kalman filtering: an optimal nonlinear framework for ensemble Kalman filters. *Mon Weath Rev.* (2012) 140:528–42. doi: 10.1175/2011MWR3640.1

27. Luo X, Moroz IM, Hoteit I. Scaled unscented transform Gaussian sum filter: theory and application. *Phys D.* (2010) 239:684–701. doi: 10.1016/j.physd.2010.01.022

28. Anderson JL, Anderson SL. A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon Weath Rev.* (1999) 127:2741–58. doi: 10.1175/1520-0493(1999)127<2741:AMCIOT>2.0.CO;2

29. Anderson JL. Spatially and temporally varying adaptive covariance inflation for ensemble filters. *Tellus.* (2009) 61A:72–83. doi: 10.1111/j.1600-0870.2008.00361.x

30. Anderson JL. Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. *Phys D.* (2007) 230:99–111. doi: 10.1016/j.physd.2006.02.011

31. Bishop CH, Hodyss D. Adaptive ensemble covariance localization in ensemble 4D-VAR state estimation. *Mon Weath Rev.* (2011) 139:1241–55. doi: 10.1175/2010MWR3403.1

32. Bocquet M. Localization and the iterative ensemble Kalman smoother. *Q J R Meteorol Soc.* (2016) 142:1075–89. doi: 10.1002/qj.2711

33. El Gharamti M. Enhanced adaptive inflation algorithm for ensemble filters. *Mon Weath Rev.* (2018) 146:623–40. doi: 10.1175/MWR-D-17-0187.1

34. Miyoshi T. The Gaussian approach to adaptive covariance inflation and its implementation with the local ensemble transform Kalman filter. *Mon Weath Rev.* (2011) 139:1519–35. doi: 10.1175/2010MWR3570.1

35. Li H, Kalnay E, Miyoshi T. Simultaneous estimation of covariance inflation and observation errors within an ensemble Kalman filter. *Q J R Meteorol Soc.* (2009) 135:523–33. doi: 10.1002/qj.371

36. Luo X, Hoteit I. Robust ensemble filtering and its relation to covariance inflation in the ensemble Kalman filter. *Mon Weath Rev.* (2011) 139:3938–53. doi: 10.1175/MWR-D-10-05068.1

37. Raanes PN, Bocquet M, Carrassi A. Adaptive covariance inflation in the ensemble Kalman filter by Gaussian scale mixtures. *Q J R Meteorol Soc.* (2019) 145:53–75. doi: 10.1002/qj.3386

38. Zhang F, Snyder C, Sun J. Impacts of initial estimate and observation availability on convective-scale data assimilation with an ensemble Kalman filter. *Mon Weath Rev.* (2004) 132:1238–53. doi: 10.1175/1520-0493(2004)132<1238:IOIEAO>2.0.CO;2

39. Dee DP. On-line estimation of error covariance parameters for atmospheric data assimilation. *Mon Weath Rev.* (1995) 123:1128–45. doi: 10.1175/1520-0493(1995)123<1128:OLEOEC>2.0.CO;2

40. Dreano D, Tandeo P, Pulido M, Ait-El-Fquih B, Chonavel T, Hoteit I. Estimating model-error covariances in nonlinear state-space models using Kalman smoothing and the expectation-maximization algorithm. *Q J R Meteorol Soc.* (2017) 143:1877–85. doi: 10.1002/qj.3048

41. Luo X. Ensemble-based kernel learning for a class of data assimilation problems with imperfect forward simulators. *PLOS ONE.* (2019) 14:e0219247. doi: 10.1371/journal.pone.0219247

42. Scheffler G, Ruiz J, Pulido M. Inference of stochastic parametrizations for model error treatment using nested ensemble Kalman filters. *Q J R Meteorol Soc.* (2019) 145:2028–45. doi: 10.1002/qj.3542

43. Luo X. Novel iterative ensemble smoothers derived from A class of generalized cost functions. *Comput Geosci.* (2021) 25:1159–89. doi: 10.1007/s10596-021-10046-1

44. Yu T, Zhu H. Hyper-parameter optimization: a review of algorithms and applications. *arXiv preprint arXiv:200305689.* (2020) doi: 10.48550/arXiv.2003.05689

45. Lindauer M, Eggensperger K, Feurer M, Biedenkapp A, Deng D, Benjamins C, et al. SMAC3: a versatile Bayesian optimization package for hyperparameter optimization. *J Mach Learn Res.* (2022) 23:54–1. doi: 10.48550/arXiv.2109.09831

46. Veloso B, Gama J, Malheiro B, Vinagre J. Hyperparameter self-tuning for data streams. *Inform Fus.* (2021) 76:75–86. doi: 10.1016/j.inffus.2021.04.011

47. Nocedal J, Wright SJ. *Numerical Optimization.* 2nd Edn. New York, NY: Springer (2006).

48. Janjić T, Nerger L, Albertella A, Schröter J, Skachko S. On domain localization in ensemble-based Kalman filter algorithms. *Mon Weath Rev.* (2011) 139:2046–60. doi: 10.1175/2011MWR3552.1

49. Fertig EJ, Hunt BR, Ott E, Szunyogh I. Assimilating non-local observations with a local ensemble Kalman filter. *Tellus A.* (2007) 59:719–30. doi: 10.1111/j.1600-0870.2007.00260.x

50. Luo X, Bhakta T. Automatic and adaptive localization for ensemble-based history matching. *J Petrol Sci Eng.* (2020) 184:106559. doi: 10.1016/j.petrol.2019.106559

51. Gaspari G, Cohn SE. Construction of correlation functions in two and three dimensions. *Q J R Meteorol Soc.* (1999) 125:723–57. doi: 10.1002/qj.49712555417

52. Ranazzi PH, Luo X, Sampaio MA. Improving pseudo-optimal Kalman-gain localization using the random shuffle method. *J Petrol Sci Eng.* (2022) 215:110589. doi: 10.1016/j.petrol.2022.110589

53. Lorenz EN, Emanuel KA. Optimal sites for supplementary weather observations: simulation with a small model. *J Atmos Sci.* (1998) 55:399–414. doi: 10.1175/1520-0469(1998)055<0399:OSFSWO>2.0.CO;2

54. Toplak M, Močnik R, Polajnar M, Bosnić Z, Carlsson L, Hasselgren C, et al. Assessment of machine learning reliability methods for quantifying the applicability domain of QSAR regression models. *J Chem Inform Model.* (2014) 54:431–41. doi: 10.1021/ci4006595

55. Burgers G, van Leeuwen PJ, Evensen G. On the analysis scheme in the ensemble Kalman filter. *Mon Weath Rev.* (1998) 126:1719–24. doi: 10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2

frontiers | Frontiers in Applied Mathematics and Statistics

TYPE Original Research
PUBLISHED 08 November 2022
DOI 10.3389/fams.2022.1014956

OPEN ACCESS

EDITED BY
Lixin Shen,
Syracuse University, United States

REVIEWED BY
Youssri Hassan Youssri,
Cairo University, Egypt
Milena Petrovic,
University of Pristina, Serbia

*CORRESPONDENCE
Fevi Novkaniza
fevi.novkaniza@sci.ui.ac.id

SPECIALTY SECTION
This article was submitted to Optimization, a section of the journal Frontiers in Applied Mathematics and Statistics

RECEIVED 09 August 2022
ACCEPTED 03 October 2022
PUBLISHED 08 November 2022

CITATION
Novkaniza F, Malik M, Sulaiman IM and Aldila D (2022) Modified spectral conjugate gradient iterative scheme for unconstrained optimization problems with application on COVID-19 model. *Front. Appl. Math. Stat.* 8:1014956. doi: 10.3389/fams.2022.1014956

Modified spectral conjugate gradient iterative scheme for unconstrained optimization problems with application on COVID-19 model

Fevi Novkaniza[1]*, Maulana Malik[1],
Ibrahim Mohammed Sulaiman[2] and Dipo Aldila[1]

[1]Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Indonesia, Depok, Indonesia, [2]School of Quantitative Sciences, Institute of Strategic Industrial Decision Modelling, Universiti Utara Malaysia, Sintok, Kedah, Malaysia

In this work, a new class of spectral conjugate gradient (CG) method is proposed for solving unconstrained optimization models. The search direction of the new method uses the ZPRP and JYJLL CG coefficients. The search direction satisfies the descent condition independent of the line search. The global convergence properties of the proposed method under the strong Wolfe line search are proved with some certain assumptions. Based on some test functions, numerical experiments are presented to show the proposed method's efficiency compared with other existing methods. The application of the proposed method for solving regression models of COVID-19 is provided.

Mathematics subject classification: 65K10, 90C52, 90C26.

KEYWORDS

unconstrained optimization, descent condition, global convergence, regression models, spectral conjugate gradient method

1. Introduction

The coronavirus disease, often called COVID-19, is an acute vector infectious disease that emerged in 2019. This disease is caused by the newly discovered coronavirus (SARS-CoV-2) and can be transmitted through droplets produced when an infected person exhales, sneezes, or coughs. Most people infected with the virus will experience mild to moderate symptoms, such as low-grade fever, runny nose, and difficulty breathing, and recover without special treatment [1].

Clinically, as of December 19, 2021, a total of 4,260,544 confirmed cases of COVID-19, with 4,111,619 recoveries and 144,002 deaths, were recorded from all regions in Indonesia since the disease was first reported in Wuhan, China [2]. To date, many studies have been carried out to model various aspects related to the coronavirus outbreak, and several researchers have also applied numerical methods to several COVID-19 models. For instance, Aggarwal et al. [3] proposed a partial differential equation model to calculate the number of COVID-19 cases in Punjab by using the modified cubic B-spline function and differential quadrature method. Other numerical methods which are applied to solve the COVID-19 model were proposed by Amar et al. [4] and Sulaiman et al. [5]. Amar et al. used various statistics and machine learning modeling approaches to forecast the COVID-19 spread in Egypt. Meanwhile, Sulaiman et al. proposed a new three-term conjugate gradient optimization method for the data from the global confirmed cases of COVID-19 from January to September 2020.

The conjugate gradient (CG) method plays an important role in solving large-scale optimization models because it uses low memory and good convergence properties. This method was first introduced by Hestenes and Stiefel [26] and is used to solve a system of linear equations. After that, in 1964, Fletcher and Reeves extended the form of the conjugate gradient method to solve large-scale nonlinear systems of equations and optimization problems without constraints. The results of the expansion carried out by Fletcher and Reeves prompted researchers to propose a new conjugate gradient method to improve computational performance and the level of convergence [6]. In 2020, Jian et al. proposed a conjugate gradient method with a spectral conjugate gradient type named the JYJLL method which is a modification of the Fletcher-Reeves (FR) and conjugate descent (CD) methods [7]. The author has determined the convergence analysis of the JYJLL method which resulted in an efficient computational performance. In addition, Zheng and Shi [8] also proposed a modification of the conjugate gradient method with a three-term type symbolized by ZPRP. This ZPRP method is an extension of the Polak-Ribiére-Polyak (PRP) method [9, 10] in which modifications are made by changing the denominator of the parameters in the PRP method. The computational performance resulting from this method is very efficient when compared to the CG-Descent method [11]. Several CG methods that have been proposed can be seen in literature [12–17]. Besides the CG method, the class of accelerated gradient descent schemes of Quasi-Newton type also contains very efficient and robust methods and can be considered for solving optimization problems. The accelerated parameters highlights can be seen in other studies [18–22]. However, in this paper we restrict the discussion to the CG method.

The CG method has recently been used to solve various problems related to optimization. For example, image reconstruction [23–25], compressed sensing [26], signal processing [27], robotic motion control [5, 15, 16, 28, 29], portfolio selection [5, 13, 14, 29–31], regression analysis [5, 32] and many more.

In this paper, we consider the general unconstrained optimization problems as follows:

$$\min_{\mathbf{r}\in\mathbb{R}^n} f(\mathbf{r}), \tag{1}$$

where $f:\mathbb{R}^n \to \mathbb{R}$ is the continuously differentiable function and its gradient is written by $\mathbf{h}(\mathbf{r}) = \nabla f(\mathbf{r})$. The iterative formula of the standard CG method can be formulated as

$$\mathbf{r}_{k+1} = \mathbf{r}_k + \alpha_k \mathbf{z}_k,\ \ k = 0, 1, 2, ... \tag{2}$$

and

$$\mathbf{z}_k := \begin{cases} -\mathbf{h}_k, & \text{if } k = 0, \\ -\mathbf{h}_k + \beta_k \mathbf{z}_{k-1}, & \text{if } k > 0, \end{cases} \tag{3}$$

where $\mathbf{r}_k$ is the current iteration, $\mathbf{h}_k$ is the gradient value of $\mathbf{h}$ at $\mathbf{r}_k$, $\mathbf{z}_k$ is the search direction, β_k is the conjugate parameter and $\alpha_k > 0$ is the step size to be obtained by some line search techniques. To calculate the step size $\alpha_k > 0$, we can use exact line search, weak Wolfe line search, or strong Wolfe line search. The exact line search is computed such that α_k satisfy

$$f(\mathbf{r}_k + \alpha_k \mathbf{z}_k) = \min f(\mathbf{r}_k + \alpha \mathbf{z}_k), \alpha > 0.$$

The weak Wolfe line search is computed such that α_k satisfy

$$f(\mathbf{r}_k + \alpha_k \mathbf{z}_k) \le f(\mathbf{r}_k) + \delta \alpha_k \mathbf{h}_k^T \mathbf{z}_k, \tag{4}$$

$$\mathbf{h}(\mathbf{r}_k + \alpha_k \mathbf{z}_k)^T \mathbf{z}_k \ge \sigma \mathbf{h}_k^T \mathbf{z}_k, \tag{5}$$

and the strong Wolfe line search is computed such that α_k satisfy

$$f(\mathbf{r}_k + \alpha_k \mathbf{z}_k) \le f(\mathbf{r}_k) + \delta \alpha_k \mathbf{h}_k^T \mathbf{z}_k,$$
$$|\mathbf{h}(\mathbf{r}_k + \alpha_k \mathbf{z}_k)^T \mathbf{z}_k| \le -\sigma \mathbf{h}_k^T \mathbf{z}_k,$$

where $0 < \delta < \sigma < 1$.

The most well-known standard CG methods are the Hestenes-Stiefel (HS) method [33], the Fletcher-Reeves (FR) method [34], the Polak-Ribiére-Polyak (PRP) method [9, 10], the Conjugate-Descent (CD) method [35], the Dai-Yuan (DY) method [36], the Liu-Storey (LS) method [37], and the Rivaie-Mustafa-Ismail-Leong (RMIL) method [38] and their β_k parameters are

$$\beta_k^{HS} = \frac{\mathbf{h}_k^T \mathbf{q}_{k-1}}{\mathbf{z}_{k-1}^T \mathbf{q}_{k-1}}, \quad \beta_k^{FR} = \frac{\|\mathbf{h}_k\|^2}{\|\mathbf{h}_{k-1}\|^2},$$
$$\beta_k^{PRP} = \frac{\mathbf{h}_k^T \mathbf{q}_{k-1}}{\|\mathbf{h}_{k-1}\|^2}, \quad \beta_k^{CD} = \frac{\|\mathbf{h}_k\|^2}{-\mathbf{z}_{k-1}^T \mathbf{h}_{k-1}},$$

$$\beta_k^{DY} = \frac{\|\mathbf{h}_k\|^2}{\mathbf{z}_{k-1}^T \mathbf{q}_{k-1}}, \quad \beta_k^{LS} = \frac{\mathbf{h}_k^T \mathbf{q}_{k-1}}{-\mathbf{h}_{k-1}^T \mathbf{z}_{k-1}}, \quad \beta_k^{RMIL} = \frac{\mathbf{h}_k^T \mathbf{q}_{k-1}}{\|\mathbf{z}_{k-1}\|^2},$$

respectively, where $\mathbf{q}_{k-1} := \mathbf{h}_k - \mathbf{h}_{k-1}$ and $\|.\|$ is a symbol for Euclidean norm on $\mathbb{R}^n$.

The $\mathbf{z}_k$ in formula (2) is the search direction used as a guide to move to the next point and must satisfy the descent direction property

$$\mathbf{h}_k^T \mathbf{z}_k < 0,\ \forall k. \tag{6}$$

It should be noted that formula (6) is an important property for the CG method to be globally convergent.

Inspired by the JYJLL method, in this study we propose a modification of the new CG method to improve the computational performance. In addition, in this study, we also apply the new method for solving a model of COVID-19 in Indonesia in which the data is taken from March 2020 (the month of the first recorded case) until May 2022.

The paper is structured as follows. In Section 2, we describe the proposed method, algorithm, and convergence analysis. In Section 3, we present the numerical experiments to show the efficiency of our new method. Finally, the application of regression models of COVID-19 using the new method is illustrated in Section 4.

2. Proposed method, algorithm and convergence analysis

Recently, Jian et al. [7] proposed a new spectral JYJLL CG method where the method satisfies the descent condition without depending on any line search. The JYJLL method is globally convergent under a weak Wolfe line search and the numerical result is efficient compared with HZ [39], KD [40], AN1 [41], and LPZ [42] methods. This new method has search direction as follows:

$$\mathbf{z}_k := \begin{cases} -\mathbf{h}_k, & \text{if } k = 0, \\ -\theta_k^{JYJLL}\mathbf{h}_k + \beta_k^{JYJLL}\mathbf{z}_{k-1}, & \text{if } k > 0, \end{cases}$$

where θ_k^{JYJLL} is the spectral parameter defined as

$$\theta_k^{JYJLL} = 1 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{-\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}, \tag{7}$$

and β_k^{JYJLL} is formulated as

$$\beta_k^{JYJLL} = \frac{\|\mathbf{h}_k\|^2 - \frac{(\mathbf{h}_k^T\mathbf{z}_{k-1})^2}{\|\mathbf{z}_{k-1}\|^2}}{\max\{\|\mathbf{h}_{k-1}\|^2, \mathbf{z}_{k-1}^T(\mathbf{h}_k - \mathbf{h}_{k-1})\}}.$$

Additionally, Zheng and Shi [8] proposed a modified three-term HS method by taking a modification to the denominator of the HS formula. The new method is named ZHS where the search direction is defined as follows:

$$\mathbf{z}_k := \begin{cases} -\mathbf{h}_k, & \text{if } k = 0, \\ -\mathbf{h}_k + \beta_k^{ZHS}\mathbf{z}_{k-1} - \beta_k^{ZHS}\frac{\mathbf{h}_k^T\mathbf{z}_{k-1}}{\mathbf{h}_k^T\mathbf{q}_{k-1}}\mathbf{q}_{k-1}, & \text{if } k > 0, \end{cases}$$

and

$$\beta_k^{ZHS} = \frac{\mathbf{h}_k^T\mathbf{q}_{k-1}}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}}, \ \mu > 0.$$

The ZHS method satisfies the sufficient descent condition without relying on a certain line search. Under some conditions, the ZHS method fulfills global convergence properties under a weak Wolfe line search and the numerical results are better than the CG-DESCENT method [39].

Motivated by the JYJLL and ZHS parameters, in this paper, the new conjugate parameter is proposed in the form as follows:

$$\beta_k^{FMSD} = \frac{\|\mathbf{h}_k\|^2 - \frac{(\mathbf{h}_k^T\mathbf{z}_{k-1})^2}{\|\mathbf{z}_{k-1}\|^2}}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}}, \ \mu > 0, \tag{8}$$

that is, replacing the JYJLL denominator with the ZHS denominator and retaining the JYJLL numerator. In addition, we retain the same formula of the spectral parameters θ_k^{FMSD} by the JYJLL method as in formula (7). So, the search direction of our proposed method is defined as follows:

$$\mathbf{z}_k := \begin{cases} -\mathbf{h}_k, & \text{if } k = 1, \\ -\theta_k^{FMSD}\mathbf{h}_k + \beta_k^{FMSD}\mathbf{z}_{k-1}, & \text{if } k > 1. \end{cases} \tag{9}$$

Our proposed method is called the spectral FMSD (Fevi-Malik-Sulaiman-Dipo) method.

Next, we give the algorithm of our proposed method below.

```
Step 1: Given any initial point $\mathbf{r}_1 \in \mathbb{R}^n$ and tolerance value $0 < \epsilon < 1$.
Step 2: Set $k = 1$, compute the gradient $\mathbf{h}_k = \nabla f(\mathbf{r}_k)$, and set $\mathbf{z}_k = -\mathbf{h}_k$.
Step 3: Compute the step length $\alpha_k$ by using any line search.
Step 4: Update point by $\mathbf{r}_{k+1} = \mathbf{r}_k + \alpha_k\mathbf{z}_k$.
Step 5: Compute $\mathbf{h}_{k+1}$. If $\|\mathbf{h}_{k+1}\| < \epsilon$, then algorithm stop. Print $\mathbf{r}^* = \mathbf{r}_{k+1}$ is best solution. Otherwise, go to the next step.
Step 6: Compute $\beta_k^{FMSD}$ by using Equation (8) and $\theta_k^{FMSD}$ by using Equation (7).
Step 7: Compute the search direction $\mathbf{z}_{k+1}$ by Equation (9).
Step 8: Go to Step 3.
```

Algorithm 1. Spectral FMSD method.

The following lemma shows that the spectral FMSD always satisfies the descent direction condition regardless of any line search.

Lemma 2.1. *Suppose that $\mathbf{z}_k$ is generated by formula (9), then*

1. *the search direction $\mathbf{z}_k$ satisfies the descent direction property, that is, $\mathbf{h}_k^T\mathbf{z}_k < 0$ for $k \geq 1$.*
2. $0 \leq \beta_k^{FMSD} \leq \frac{\mathbf{h}_k^T\mathbf{z}_k}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}$.

Proof: We will prove the theorem by induction. For $k = 1$, it is true, i.e., $\mathbf{h}_1^T\mathbf{z}_1 = \|\mathbf{h}_1\|^2$. Now, assume that $\mathbf{h}_{k-1}^T\mathbf{z}_{k-1} < 0$ is true for $k - 1$, thus we prove $\mathbf{h}_k^T\mathbf{z}_k < 0$ is true for k. With regard to formula (8), the proof is divided into two cases, as presented below:

- **Case 1**: if $\mathbf{z}_{k-1}^T\mathbf{q}_{k-1} \leq \mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|$ and $\mu > 0$, then

$$\begin{aligned}\mathbf{z}_{k-1}^T\mathbf{q}_{k-1} &= \mathbf{z}_{k-1}^T(\mathbf{h}_k - \mathbf{h}_{k-1})\\ &= \mathbf{h}_k^T\mathbf{z}_{k-1} - \mathbf{h}_{k-1}^T\mathbf{z}_{k-1} \leq \mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|,\end{aligned}$$

it implies

$$\mathbf{h}_k^T\mathbf{z}_{k-1} \leq \mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\| + \mathbf{h}_{k-1}^T\mathbf{z}_{k-1}. \tag{10}$$

Let θ_k is angle between $\mathbf{h}_k$ and $\mathbf{z}_{k-1}$, then

$$\cos\theta_k = \frac{\mathbf{h}_k^T\mathbf{z}_{k-1}}{\|\mathbf{h}_k\|\|\mathbf{z}_{k-1}\|}. \tag{11}$$

From formulas (8), (7), (9), (10) and (11), we have

$$\begin{aligned}&\mathbf{h}_k^T\mathbf{z}_k\\ &= \mathbf{h}_k^T(-\theta_k^{FMSD}\mathbf{h}_k + \beta_k^{FMSD}\mathbf{z}_{k-1})\\ &= -\theta_k^{FMSD}\|\mathbf{h}_k\|^2 + \beta_k^{FMSD}\mathbf{h}_k^T\mathbf{z}_{k-1}\\ &= -\left[1 - \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right]\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \frac{(\mathbf{h}_k^T\mathbf{z}_{k-1})^2}{\|\mathbf{z}_{k-1}\|^2}}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\mathbf{h}_k^T\mathbf{z}_{k-1}\\ &= -\|\mathbf{h}_k\|^2 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\mathbf{h}_k^T\mathbf{z}_{k-1}\\ &\leq -\|\mathbf{h}_k\|^2 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\\ &\times(\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\| + \mathbf{h}_{k-1}^T\mathbf{z}_{k-1})\\ &= \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2(1-\cos^2\theta_k)}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}\\ &-\|\mathbf{h}_k\|^2\cos^2\theta_k \leq \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2\sin^2\theta_k}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\\ &\mathbf{h}_{k-1}^T\mathbf{z}_{k-1} < 0.\end{aligned} \tag{12}$$

- **Case 2**: if $\mathbf{z}_{k-1}^T\mathbf{q}_{k-1} > \mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|$ and $\mu > 0$, then $\mathbf{z}_{k-1}^T\mathbf{q}_{k-1} > 0$. Using formulas (8), (7), (9), and (11), we get

$$\begin{aligned}\mathbf{h}_k^T\mathbf{z}_k &= \mathbf{h}_k^T(-\theta_k^{FMSD}\mathbf{h}_k + \beta_k^{FMSD}\mathbf{z}_{k-1})\\ &= -\theta_k^{FMSD}\|\mathbf{h}_k\|^2 + \beta_k^{FMSD}\mathbf{h}_k^T\mathbf{z}_{k-1}\\ &= -\left[1 - \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right]\|\mathbf{h}_k\|^2\\ &\quad + \frac{\|\mathbf{h}_k\|^2 - \frac{(\mathbf{h}_k^T\mathbf{z}_{k-1})^2}{\|\mathbf{z}_{k-1}\|^2}}{\mathbf{z}_{k-1}^T\mathbf{q}_{k-1}}\mathbf{h}_k^T\mathbf{z}_{k-1}\end{aligned}$$

$$\begin{aligned}&= -\|\mathbf{h}_k\|^2 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\mathbf{z}_{k-1}^T\mathbf{q}_{k-1}}\\ &\quad\times \mathbf{h}_k^T\mathbf{z}_{k-1}\\ &= -\|\mathbf{h}_k\|^2 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\mathbf{z}_{k-1}^T\mathbf{q}_{k-1}}\\ &\quad\times (\mathbf{h}_k^T\mathbf{z}_{k-1} - \mathbf{h}_{k-1}^T\mathbf{z}_{k-1} + \mathbf{h}_{k-1}^T\mathbf{z}_{k-1})\\ &= -\|\mathbf{h}_k\|^2 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\mathbf{z}_{k-1}^T\mathbf{q}_{k-1}}\\ &\quad\times (\mathbf{z}_{k-1}^T\mathbf{q}_{k-1} + \mathbf{h}_{k-1}^T\mathbf{z}_{k-1})\\ &= \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2(1-\cos^2\theta_k)}{\mathbf{z}_{k-1}^T\mathbf{q}_{k-1}}\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}\\ &\quad - \|\mathbf{h}_k\|^2\cos^2\theta_k \qquad (13)\\ &= \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2\sin^2\theta_k}{\mathbf{z}_{k-1}^T\mathbf{q}_{k-1}}\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}\\ &\quad - \|\mathbf{h}_k\|^2\cos^2\theta_k\\ &\leq \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2\sin^2\theta_k}{\mathbf{z}_{k-1}^T\mathbf{q}_{k-1}}\mathbf{h}_{k-1}^T\mathbf{z}_{k-1} < 0.\end{aligned}$$

Hence, $\mathbf{h}_k^T\mathbf{z}_k < 0$ is satisfied for $k \geq 1$.

Next, we will prove the interval of β_k^{FMSD}. From formulas (12 and (13, we obtain the relation $\mathbf{h}_k^T\mathbf{z}_k \leq \beta_k^{FMSD}\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}$. Furthermore, since $\mathbf{h}_k^T\mathbf{z}_k < 0$, we have $\beta_k^{FMSD} \leq \frac{\mathbf{h}_k^T\mathbf{z}_k}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}$.

Now, from formulas (8) and (7), we get

$$\begin{aligned}\beta_k^{FMSD} &= \frac{\|\mathbf{h}_k\|^2 - \frac{(\mathbf{h}_k^T\mathbf{z}_{k-1})^2}{\|\mathbf{z}_{k-1}\|^2}}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}}\\ &= \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}}\\ &= \frac{\|\mathbf{h}_k\|^2\sin^2\theta_k}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}} \geq 0.\end{aligned}$$

Thus, $0 \leq \beta_k^{FMSD} \leq \frac{\mathbf{h}_k^T\mathbf{z}_k}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}$ holds. The proof is complete.

In the analysis below, we establish the global convergence properties of the spectral FMSD method. First, we need the following assumption, proposition, and Zoutendijk conditions.

Assumption 2.2. *(A1) The level set $\mathcal{B}:=\{\mathbf{r} \in \mathbb{R}^n : f(\mathbf{r}) \leq f(\mathbf{r}_0)\}$ is bounded where $\mathbf{r}_0$ is the starting point; (A2) In a neighborhood $\mathcal{L}$ of $\mathcal{B}$ the function f is continuously differentiable and its gradient Lipschitz continuous on $\mathcal{H}$. That is, we can find $L > 0$ such that*

$$\|\mathbf{h}(\mathbf{r}) - \mathbf{h}(\mathbf{s})\| \leq L\|\mathbf{r} - \mathbf{s}\|, \ \forall \mathbf{r}, \mathbf{s} \in \mathcal{L}.$$

Proposition 2.3. *Suppose that* $\mathbf{z}_k$ *is generated by formula (9) and Assumption 2.2 holds. If the step length* α_k *is calculated by weak Wolfe line search (4) and (5), then*

$$\alpha_k \geq \frac{(\sigma - 1)\mathbf{h}_k^T\mathbf{z}_k}{L\|\mathbf{z}_k\|^2}, \tag{14}$$

where σ *and* L *are positive constant in Assumption 2.2 and formula (5) respectively.*

Proof: Both sides of formula (5) are subtracted by $\mathbf{h}_k^T\mathbf{z}_k$, we get

$$(\sigma - 1)\mathbf{h}_k^T\mathbf{z}_k \leq (\mathbf{h}_{k+1} - \mathbf{h}_k)^T\mathbf{z}_k = \mathbf{q}_k^T\mathbf{z}_k \leq \|\mathbf{q}_k\|\|\mathbf{z}_k\|,$$

combining with Lipschitz continuity, we obtain

$$(\sigma - 1)\mathbf{h}_k^T\mathbf{z}_k \leq \alpha_k L\|\mathbf{z}_k\|^2.$$

Since $\mathbf{z}_k$ is a descent direction and $\sigma < 1$, formula (14 holds immediately.

Zoutendijk condition [43] is often used to prove the global convergence of the CG method. The following lemma shows that the Zoutendijk condition holds for the proposed method under the weak Wolfe line search conditions formulas (4) and (5).

Lemma 2.4. *Suppose Assumption 2.2 holds and consider any iterative expression formula (2, where* $\mathbf{z}_k$ *is generated by formula formula (9). If* α_k *is calculated by weak Wolfe line search formulas (4) and (5), then the following so-called Zoutendijk condition holds:*

$$\sum_{k=1}^{\infty} \frac{(\mathbf{h}_k^T\mathbf{z}_k)^2}{\|\mathbf{z}_k\|^2} < +\infty. \tag{15}$$

Proof: From weak Wolfe condition (4), we have

$$f(\mathbf{r}_k) - f(\mathbf{r}_k + \alpha_k\mathbf{z}_k) \geq -\delta\alpha_k\mathbf{h}_k^T\mathbf{z}_k,$$

combining with formula (14), we get

$$f(\mathbf{r}_k) - f(\mathbf{r}_k + \alpha_k\mathbf{z}_k) \geq \frac{\delta(1-\sigma)(\mathbf{h}_k^T\mathbf{z}_k)^2}{L\|\mathbf{z}_k\|^2}. \tag{16}$$

Summing up both sides of formula (16), and applying the condition (A1) in Assumption 2.2, zoutendijk condition (15) holds.

Lemma 2.5. *Suppose that Assumption 2.2 holds and consider the sequences* $\{\mathbf{h}_k\}$ *and* $\{\mathbf{z}_k\}$ *are generated by Algorithm 1, where* α_k *is calculated by weak Wolfe line search (4–(5, then*

$$\frac{\|\mathbf{z}_k\|^2}{(\mathbf{h}_k^T\mathbf{z}_k)^2} \leq \sum_{i=1}^{k} \frac{1}{\|\mathbf{h}_i\|^2}. \tag{17}$$

Proof: From formula (9), we have

$$\mathbf{z}_k + \theta_k^{FMSD}\mathbf{h}_k = \beta_k^{FMSD}\mathbf{z}_{k-1}. \tag{18}$$

Squaring up both sides of formula (18) and using the first condition in Lemma 2.1, we obtain

$$\begin{aligned}\|\mathbf{z}_k\|^2 &= (\beta_k^{FMSD})^2\|\mathbf{z}_{k-1}\|^2 - 2\theta_k^{FMSD}\mathbf{h}_k^T\mathbf{z}_k - (\theta_k^{FMSD})^2\|\mathbf{h}_k\|^2 \\ &\leq \left(\frac{\mathbf{h}_k^T\mathbf{z}_k}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right)^2 \|\mathbf{z}_{k-1}\|^2 - 2\theta_k^{FMSD}\mathbf{h}_k^T\mathbf{z}_k \\ &\quad - (\theta_k^{FMSD})^2\|\mathbf{h}_k\|^2,\end{aligned}$$

multiplying up both sides by $\frac{1}{(\mathbf{h}_k^T\mathbf{z}_k)^2}$, we get

$$\begin{aligned}\frac{\|\mathbf{z}_k\|^2}{(\mathbf{h}_k^T\mathbf{z}_k)^2} &\leq \left(\frac{\|\mathbf{z}_{k-1}\|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right)^2 - \frac{2\theta_k^{FMSD}}{\mathbf{h}_k^T\mathbf{z}_k} - \frac{(\theta_k^{FMSD})^2\|\mathbf{h}_k\|^2}{(\mathbf{h}_k^T\mathbf{z}_k)^2} \\ &= \left(\frac{\|\mathbf{z}_{k-1}\|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right)^2 - \left(\frac{1}{\|\mathbf{h}_k\|} + \frac{\theta_k^{FMSD}\|\mathbf{h}_k\|}{\mathbf{h}_k^T\mathbf{z}_k}\right)^2 + \frac{1}{\|\mathbf{h}_k\|^2} \\ &\leq \left(\frac{\|\mathbf{z}_{k-1}\|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right)^2 + \frac{1}{\|\mathbf{h}_k\|^2}.\end{aligned}$$

Since $\mathbf{z}_1 = -\mathbf{h}_1$ holds, we obtain

$$\begin{aligned}\frac{\|\mathbf{z}_k\|^2}{(\mathbf{h}_k^T\mathbf{z}_k)^2} &\leq \left(\frac{\|\mathbf{z}_{k-1}\|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right)^2 + \frac{1}{\|\mathbf{h}_k\|^2} \\ &\leq \left(\frac{\|\mathbf{z}_{k-2}\|}{\mathbf{h}_{k-2}^T\mathbf{z}_{k-2}}\right)^2 + \frac{1}{\|\mathbf{h}_{k-1}\|^2} + \frac{1}{\|\mathbf{h}_k\|^2} \\ &\leq \left(\frac{\|\mathbf{z}_{k-3}\|}{\mathbf{h}_{k-3}^T\mathbf{z}_{k-3}}\right)^2 + \frac{1}{\|\mathbf{h}_{k-2}\|^2} + \frac{1}{\|\mathbf{h}_{k-1}\|^2} + \frac{1}{\|\mathbf{h}_k\|^2} \\ &\leq \ldots \leq \sum_{i=1}^{k} \frac{1}{\|\mathbf{h}_i\|^2}.\end{aligned}$$

The proof is finished.

Based on Lemmas 2.1, 2.4, and 2.5, we can establish the theorem of global convergence of the FMSD method.

Theorem 2.6. *Suppose that Assumption 2.2 is satisfied. Consider* $\{\mathbf{r}_k\}$ *is generated by Algorithm 1, where* α_k *is calculated by weak Wolfe line search (4–5), then*

$$\lim_{k\to\infty} \inf \|\mathbf{h}_k\| = 0. \tag{19}$$

Proof: We prove this theorem by contradiction. Suppose that formula (19) is not true, then there exists a positive constant $a > 0$ such that

$$\|\mathbf{h}_k\| \geq a, \forall k \geq 1.$$

Using the above relation and formula (17), we obtain

$$\frac{\|\mathbf{z}_k\|^2}{(\mathbf{h}_k^T\mathbf{z}_k)^2} \leq \sum_{i=1}^{k} \frac{1}{\|\mathbf{h}_i\|^2} \leq \frac{k}{a^2}.$$

It implies

$$\sum_{k=1}^{\infty} \frac{(\mathbf{h}_k^T\mathbf{z}_k)^2}{\|\mathbf{z}_k\|^2} \geq \sum_{k=1}^{\infty} \frac{a^2}{k} = +\infty,$$

which contradicts with the Zoutendijk condition in formula (15). Hence, formula (19) is true. The proof is finished.

3. Numerical experiments

In this part, we report the numerical experiments of the FMSD method and compare the computational performance with the JYJLL method proposed by Jian et al. [7]. Both the methods were coded in MATLAB 2019a and ran using a personal computer with an Intel Core i7 processor, 16 GB RAM, 64 bit Windows 10 Pro operating system. The comparisons are made under the weak Wolfe line search (4–5) with $\sigma = 0.2$ and $\delta = 0.02$ for the FMSD method and $\sigma = 0.1$ and $\delta = 0.01$ for the JYJLL method. We tested 132 unconstrained problems in the CUTEr library suggested by Andrei [6, 44] and Moré et al. [45] with dimensions from 2 to 1,000,000. Mostly, we used two different dimensions for the problem and the iteration stopped using the $\|\mathbf{h}_k\|_\infty \leq 10^{-6}$ criteria. The initial point used for all problems can be seen in Jiang et al. [25]. Table 1 details the test function and dimensions of the test problems.

Detailed numerical results are provided in Table 2 which include the number of iterations (NOI), the total number of function evaluations (NOF), and the CPU time in seconds (CPU). In Table 2, "-" indicates that the methods failed to solve the corresponding problems within 2000 iterations.

To clearly determine a method that has good computational performance, here we use the performance profiles suggested by Dolan and Moré [46] to show the performance under NOI, NOF, and CPU time, respectively. Comparison results are obtained by running a solver on a set P of problems and recording relevant information such as NOI, NOF, and CPU time. Suppose that S is the set of solvers under consideration and assume S is made up of n_s solvers and P is made up of n_p problems. For each problem $p \in P$ and solver $s \in S$, we denote $t_{p,s}$ as the CPU time (or NOI or NOF, etc.) required to solve problem $p \in P$ by solver $s \in S$. The comparison between different solvers is based on the performance ratio described by

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}}.$$

Let $\rho_s(\tau)$ be the probability for solver $s \in S$ that a performance ratio $r_{p,s}$ is within a factor $\tau \in \mathbb{R}^n$. For example, the value of

TABLE 1 The problems and their dimensions.

No	Problem/Dimension	No	Problem/Dimension
1	COSINE 6,000	67	Extended DENSCHNB 300,000
2	COSINE 100,000	68	Generalized Quartic 9,000
3	COSINE 800,000	69	Generalized Quartic 90,000
4	DIXMAANA 2,000	70	Generalized Quartic 500,000
5	DIXMAANA 30,000	71	BIGGSB1 110
6	DIXMAANB 8,000	72	BIGGSB1 200
7	DIXMAANB 16,000	73	SINE 100,000
8	DIXMAANC 900	74	SINE 50,000
9	DIXMAANC 9,000	75	FLETCBV 15
10	DIXMAAND 4,000	76	FLETCBV 55
11	DIXMAAND 30,000	77	NONSCOMP 5,000
12	DIXMAANE 800	78	NONSCOMP 80,000
13	DIXMAANE 16,000	79	POWER 150
14	DIXMAANF 5,000	80	POWER 90
15	DIXMAANF 20,000	81	RAYDAN1 500
16	DIXMAANG 4,000	82	RAYDAN1 5,000
17	DIXMAANG 30,000	83	RAYDAN2 2,000
18	DIXMAANH 2,000	84	RAYDAN2 20,000
19	DIXMAANH 50,000	85	RAYDAN2 500,000
20	DIXMAANI 120	86	DIAGONAL1 800
21	DIXMAANI 12	87	DIAGONAL1 2,000
22	DIXMAANJ 1,000	88	DIAGONAL2 100
23	DIXMAANJ 5,000	89	DIAGONAL2 1,000
24	DIXMAANK 4,000	90	DIAGONAL3 500
25	DIXMAANK 40	91	DIAGONAL3 2,000
26	DIXMAANL 800	92	Discrete Boundary Value 2,000
27	DIXMAANL 8,000	93	Discrete Boundary Value 20,000
28	DIXON3DQ 150	94	Discrete Integral Equation 500
29	DIXON3DQ 15	95	Discrete Integral Equation 1,500
30	DQDRTIC 9,000	96	Extended Powell Singular 1,000
31	DQDRTIC 90,000	97	Extended Powell Singular 2,000
32	QUARTICM 5000	98	Linear Full Rank 100
33	QUARTICM 150,000	99	Linear Full Rank 500
34	EDENSCH 7,000	100	Osborne 2 11
35	EDENSCH 40,000	101	Penalty1 200
36	EDENSCH 500,000	102	Penalty1 1,000
37	EG2 100	103	Penalty2 100
38	EG2 35	104	Penalty2 110
39	FLETCHCR 1,000	105	Extended Rosenbrock 500
40	FLETCHCR 50,000	106	Extended Rosenbrock 1,000
41	FLETCHCR 200,000	107	Broyden Tridiagonal 500
42	Freudenstein and Roth 460	108	Broyden Tridiagonal 50
43	Freudenstein and Roth 10	109	HIMMELH 70,000
44	Generalized Rosenbrock 10,000	110	HIMMELH 240,000

(Continued)

TABLE 1 (Continued)

No	Problem/dimension	No	Problem/dimension
45	Generalized Rosenbrock 100	111	Brown Badly Scaled 2
46	HIMMELBG 70,000	112	Brown and Dennis 4
47	HIMMELBG 240,000	113	Biggs EXP6 6
48	LIARWHD 15	114	Osborne1 5
49	LIARWHD 1,000	115	Extended Beale 5,000
50	Extended Penalty 1,000	116	Extended Beale 10,000
51	Extended Penalty 8,000	117	HIMMELBC 500,000
52	QUARTC 4,000	118	HIMMELBC 1,000,000
53	QUARTC 80,000	119	ARWHEAD 100
54	QUARTC 500,000	120	ARWHEAD 1,000
55	TRIDIA 300	121	ENGVAL1 500,000
56	TRIDIA 50	122	ENGVAL1 1,000,000
57	Extended Woods 150,000	123	DENSCHNA 500,000
58	Extended Woods 200,000	124	DENSCHNA 1,000,000
59	BDEXP 5,000	125	DENSCHNB 500,000
60	BDEXP 50,000	126	DENSCHNB 1,000,000
61	BDEXP 500,000	127	DENSCHNC 10
62	DENSCHNF 90,000	128	DENSCHNC 500
63	DENSCHNF 280,000	129	DENSCHNF 500,000
64	DENSCHNF 600,000	130	DENSCHNF 1,000,000
65	DENSCHNB 6,000	131	ENGVAL8 500,000
66	DENSCHNB 24,000	132	ENGVAL8 1,000,000

$\rho_s(1)$ is the probability that the solver will win over the rest of the solvers. The formula of $\rho_s(\tau)$ is defined as follows:

$$\rho_s(\tau) = \frac{1}{n_p} size\{p \in P : \log r_{p,s} \leq \tau\}.$$

According to the rule of the performance profile above, we can describe the performance curves based on Table 2 as in Figures 1–3. Based on the three figures, we can see that the FMSD method is superior to the JYJLL method under the unconstrained problems in Table 1.

4. Application to regression models of COVID-19

SARS-CoV-2 virus popularly known as the COVID-19 infection was first reported in the Asian continent from Wuhan province, Hubei city of China toward the end of 2019. As of 20 June 2022, almost all the countries in Asia except Turkmenistan have reported at least one case of the infection [47]. However, countries that include India, South Korea, Vietnam, Japan, and Iran recorded the highest rates of confirmed cases of the

TABLE 2 Numerical results.

No	JYJLL			FMSD		
	NOI	NOF	CPU	NOI	NOF	CPU
1	33	103	0.1355	30	94	0.0694
2	184	374	2.8852	126	258	1.8144
3	55	170	10.2971	43	155	8.256
4	20	83	0.2293	17	80	0.1863
5	20	89	2.9297	20	92	2.7478
6	24	93	1.0052	22	91	0.7986
7	24	93	1.6604	25	87	1.4167
8	25	89	0.2524	24	87	0.1867
9	10	73	0.7633	15	79	0.7799
10	21	90	0.4765	21	99	0.4898
11	21	87	2.7908	16	89	2.6687
12	-	-	-	1,400	2,385	2.2021
13	-	-	-	1,028	1,827	2.1568
14	1,535	2,679	2.3565	744	1,317	1.1824
15	1,887	3,232	3.3751	753	1,312	1.4909
16	-	-	-	1,029	1,755	1.5518
17	-	-	-	1,081	1,815	2.0251
18	-	-	-	795	1,372	1.2811
19	-	-	-	-	-	-
20	-	-	-	-	-	-
21	1,465	2,479	0.6701	882	1,537	0.0938
22	-	-	-	-	-	-
23	1,241	2,239	15.5235	-	-	-
24	829	1,490	7.6136	515	918	4.8861
25	-	-	-	1,172	1,994	0.202
26	-	-	-	1,899	3,277	3.0415
27	853	1,539	14.538	665	1,178	11.0235
28	-	-	-	-	-	-
29	422	740	0.2037	472	800	0.0249
30	446	822	0.2846	343	646	0.2215
31	414	776	2.0585	301	571	0.878
32	42	143	0.2467	38	143	0.2453
33	84	250	11.6764	110	317	15.0269
34	43	165	0.3478	30	109	0.2601
35	44	192	2.3017	69	421	5.0274
36	85	603	89.6504	121	954	142.2274
37	-	-	-	-	-	-
38	-	-	-	-	-	-
39	116	207	0.0085	77	158	0.0066
40	109	235	0.2867	120	278	0.329
41	321	3,144	13.5325	101	777	3.0531
42	-	-	-	1,814	7,922	0.2898
43	-	-	-	1,334	2,893	0.0774
44	-	-	-	-	-	-
45	-	-	-	-	-	-
46	2	15	0.0425	2	16	0.0477

(Continued)

TABLE 2 (Continued)

No	JYJLL			FMSD		
	NOI	NOF	CPU	NOI	NOF	CPU
47	2	13	0.6799	2	13	0.1117
48	117	219	0.008	74	163	0.0051
49	836	1,428	0.0472	607	1,063	0.0383
50	32	212	0.3311	20	123	0.1931
51	16	93	10.9861	16	93	10.9116
52	50	156	0.1896	46	149	0.1825
53	89	263	6.926	73	228	5.5093
54	109	323	51.3058	114	346	53.7087
55	-	-	-	-	-	-
56	1,187	2028	0.2817	508	917	0.0253
57	-	-	-	1,426	2,464	9.321
58	-	-	-	-	-	-
59	2	11	0.007	2	12	0.0065
60	2	16	0.0592	2	10	0.0456
61	2	12	2.1566	2	13	0.564
62	24	102	0.154	24	102	0.1715
63	23	106	0.5798	23	106	0.6301
64	25	111	1.5076	30	115	1.309
65	21	86	0.0093	18	79	0.008
66	21	90	0.0416	20	88	0.0446
67	19	91	1.4781	19	91	0.4131
68	19	83	0.0314	18	82	0.0296
69	16	82	0.1376	17	87	0.1665
70	25	101	1.1407	17	89	0.8401
71	1,684	2,890	0.0903	809	1,373	0.0384
72	-	-	-	1,350	2,316	0.0675
73	61	192	1.4482	44	168	1.2826
74	59	168	0.9694	28	98	0.426
75	91	195	0.0065	67	114	0.0049
76	1,425	2,148	1.066	1,707	2,213	0.0983
77	45	118	0.0164	45	118	0.0216
78	87	203	0.5009	81	184	0.2934
79	239	423	0.0113	127	228	0.0103
80	1,817	3,102	0.0849	733	1,306	0.0503
81	713	1,266	0.0438	636	1,094	0.0428
82	-	-	-	-	-	-
83	14	71	0.0054	12	67	0.0055
84	15	97	0.1103	17	105	0.1101
85	37	251	3.8983	82	592	8.081
86	904	4,004	0.2162	700	4,172	0.2266
87	-	-	-	1,357	6,469	0.6286
88	133	265	0.0098	164	294	0.012
89	962	1,696	0.231	748	1,327	0.0893
90	1,068	3,825	0.2293	906	4,470	0.3026
91	-	-	-	1,818	10,110	1.6853
92	232	425	6.5431	105	200	2.4856

(Continued)

TABLE 2 (Continued)

No	JYJLL			FMSD		
	NOI	NOF	CPU	NOI	NOF	CPU
93	0	0	1.0216	0	0	1.0422
94	13	59	4.3587	13	59	4.4905
95	16	63	43.3358	18	64	44.5679
96	-	-	-	-	-	-
97	-	-	-	-	-	-
98	13	63	0.0846	15	71	0.0414
99	18	84	0.4085	18	84	0.3522
100	-	-	-	-	-	-
101	-	-	-	1,400	2,966	0.4322
102	-	-	-	1,268	2,517	19.2961
103	337	880	0.1366	310	746	0.1023
104	202	576	0.2837	176	648	0.0872
105	-	-	-	-	-	-
106	-	-	-	-	-	-
107	52	115	0.1412	54	116	0.1451
108	40	95	0.1788	40	95	0.0091
109	33	155	0.952	-	-	-
110	23	105	2.2911	32	185	3.6654
111	-	-	-	-	-	-
112	266	1,125	0.0659	271	1,272	0.043
113	-	-	-	-	-	-
114	-	-	-	-	-	-
115	369	647	0.8829	344	583	0.8067
116	-	-	-	418	750	2.253
117	29	110	0.8918	27	102	0.8617
118	28	108	1.713	29	110	1.7295
119	25	74	0.003	22	70	0.0033
120	27	99	0.0063	27	85	0.0038
121	113	1,048	10.5062	86	754	6.7726
122	182	1,647	29.8122	101	836	14.5085
123	37	112	9.4347	36	108	8.7515
124	35	109	17.9446	35	109	18.1365
125	27	106	0.7703	26	108	0.8496
126	23	111	1.5522	28	116	1.6992
127	38	128	0.0045	34	110	0.0061
128	62	170	0.0254	41	127	0.0195
129	343	1,161	12.0093	531	1,857	18.929
130	387	1,366	28.4407	400	1,441	28.0926
131	263	2,557	24.3792	168	1,589	14.9358
132	466	4,731	88.6258	175	1,608	28.5672

infection [48]. The first positive COVID-19 case in Indonesia was recorded on March 2, 2020, but within the first 6 weeks, the presence of the virus has been confirmed in almost all the provinces of the country [49]. Despite the early wide-scale response from the government, the country has recorded a high

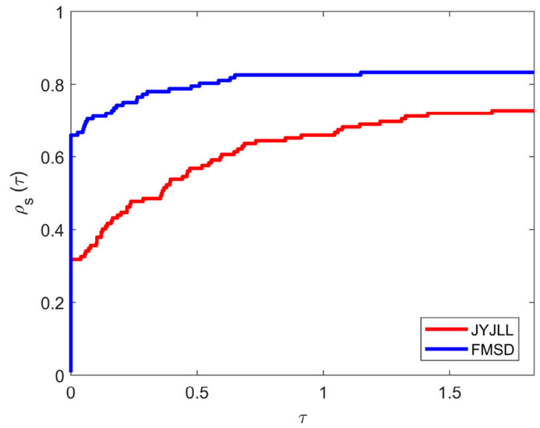


FIGURE 1
Performance profiles on NOI.

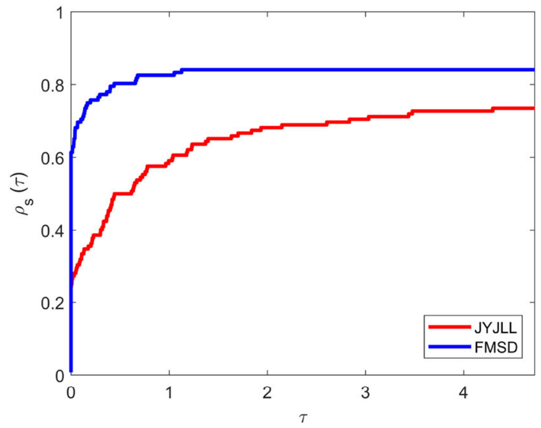


FIGURE 3
Performance profiles on CPU Time.

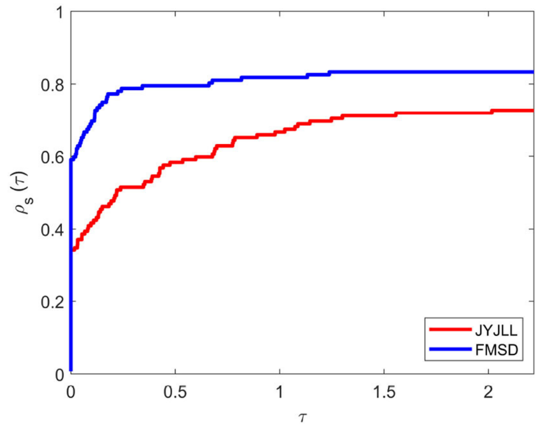


FIGURE 2
Performance profiles on NOF.

number of deaths from the positive cases of the infection [50]. According to the WHO, Indonesia has so far recorded a total of 156,695 deaths from a total of 6,069,255 confirmed cases of the infection as of 20 June 2022 [51] of which more than 750 deaths are front-line health workers. Based on recent figures, we can say that Indonesia has been able to contain the disease outbreak. This can be attributed to the admirable resilience of the country's front-line health workers, strict health protocols, and successful vaccination programs. Data from the WHO shows that the total of people that have been administered the vaccine doses as of 15 June 2020 stands at 417,522,347 [51].

In recent times, several works of literature have employed different mathematical and numerical approaches for modeling the COVID-19 outbreak [see [5, 32, 52]]. This paper aims to study the performance of the proposed method on a parameterized COVID-19 regression model. For deriving the COVID-19 regression model, the study will consider the total Indonesian monthly positive confirmed cases of the infection from March 2020 (the month of the first recorded case) until May 2022. The obtained data would be transformed into an unconstrained optimization model which would later be solved using the proposed method.

A regression analysis function of the form:

$$y = h(x_1, x_2, ..., x_p + \varepsilon), \tag{20}$$

has the response variable denoted by y, ε represents the error, and the predictor is given as x_i, $i = 1, 2, ..., p$, $p > 0$. The type of function plays an important role in the statistical modeling of problems in applied sciences, physical sciences, management sciences, and more. Based on the above description, we can describe regression analysis as a statistical procedure employed to estimate the relationships between a dependent and one or several independent variables. For any given regression analysis-related problem, the linear regression function can be derived by computing y such that

$$y = a_0 + a_1x_1 + a_2x_2 + \ldots + a_px_p + \varepsilon \tag{21}$$

with $a_0, \ldots, a_p$ representing the regression parameters. These parameters $a_0, a_1, \ldots, a_p$ are estimated to minimize the error ε value. Based on several works of literature, the linear regression process rarely occurs in situations because most problems are often nonlinear in nature. Based on the non-linearity of the problems, studies usually consider the nonlinear regression process [5]. This and other considerations motivated the idea of using the nonlinear regression procedure in this study.

To construct the parameterized regression model, we considered the death cases recorded from those infected by the COVID-19 virus from the first month Indonesia confirmed the first case; March 2020 until May 2022, totaling 27 months.

TABLE 3 Statistics of confirmed positive cases and death recorded from COVID-19 infection in Indonesia from March 2020 to May, 2022.

Monthly data (Mar 2020 – May 2022) (x)	Total confirmed Cases per month (y)	Total death Per month
1	1,528	136
2	8,590	656
3	16,355	821
4	29,912	1,263
5	51,991	2,255
6	66,360	2,212
7	112,212	3,397
8	123,080	3,129
9	128,795	3,076
10	204,315	5,193
11	335,116	7,860
12	256,320	6,168
13	177,078	4,692
14	156,656	4,663
15	156,335	5,057
16	356,569	7,913
17	1,231,386	35,628
18	680,143	38,372
19	125,303	9,448
20	29,254	1,466
21	12,051	425
22	6,311	258
23	90,650	232
24	1,211,078	4,015
25	448,379	6,754
26	33,978	1,166
27	8,177	334

The data were obtained from the *Indonesia COVID Coronavirus Statistics Worldometer* [53] and the detailed description of the model formulation process was presented as follows. Note: it may be confirmed that the statistics of recorded cases are less than the actual number, this might be a result of limited testing. From the data presented in Table 3, the x-variable would represent the months considered while the y-variable represents the confirmed death cases for that month. Also, only data of 26 months (March 2020 to April 2022) would be considered for data fitting because data for May 2022 would be reserved for error analysis.

Based on the data of x and y given in Table 3, the approximate function for the nonlinear least square method was obtained as follows:

$$f(x) = -842.24 + 35865.66x - 909.17x^2. \tag{22}$$

TABLE 4 Performance results of FMSD method for optimization of the quadratic model Equation (25).

Initial points	NOI	CPU time
(0.5,0.5,0.5)	13	0.11176541011740658
(5,5,5)	16	0.04775448062163305
(11,11,11)	17	0.84012854607846890

The above function (22) will be utilized when approximating the y data values based on x data values. Since this study considered the monthly confirmed cases, the x_j would be used to denote the months while y_j will present the confirmed cases for that month. Based on this information, the least squares method defined by function (22) would be transformed into an unconstrained minimization problem of the form:

$$\min_{x\in\mathbb{R}^n} f(x) = \sum_{j=1}^{n}\left(\left(u_0 + u_1x_j + u_2x_j^2\right) - y_j\right)^2. \tag{23}$$

The data for the first 26 months from Table 3 will be used to derive the nonlinear quadratic function for the least square method. The derived function would be extended to construct the unconstrained optimization function. Based on the above discussion, it is obvious that there exist some parabolic relations between the regression parameters u_0, u_1, u_2, the regression function (20) with the data x_j and the value of y_j.

$$\min_{x\in\mathbb{R}^2} \sum_{j=1}^{n} E_j^2 = \sum_{j=1}^{n}\left(\left(u_0 + u_1x_j + u_2x_j^2\right) - y_j\right)^2. \tag{24}$$

To define the nonlinear quadratic unconstrained minimization model, Equation (24 would be transformed using data from Table 3 as follows:

$$\begin{aligned}&26u_1^2 + 702u_1u_2 + 12402u_1u_3 - 209806038u_1 + 2610621u_2^2\\&+ 246402u_2u_3 - 17172778u_2 + 15333u_3^2\\&- 4006838782u_3 + 4152673772991.\end{aligned} \tag{25}$$

The above nonlinear quadratic model was constructed using data from the first month until the 26th month because the data for the 27th month was reserved for relative error analysis of the predicted data. Now, we can apply the proposed method to solve the model (25). The results presented in Table 4 illustrate the performance of the proposed FMSD algorithm for problem (25) under the weak Wolfe line search conditions (4–5).

The proposed method was employed as an alternative method to compute the values of u_0, u_1, u_2 because of the difficulty faced when using matrix inverse. For the proposed method, different initial points were considered for the model. The iteration was terminated if the iterations exceeded 1,000 or the method was unable to solve the problem.

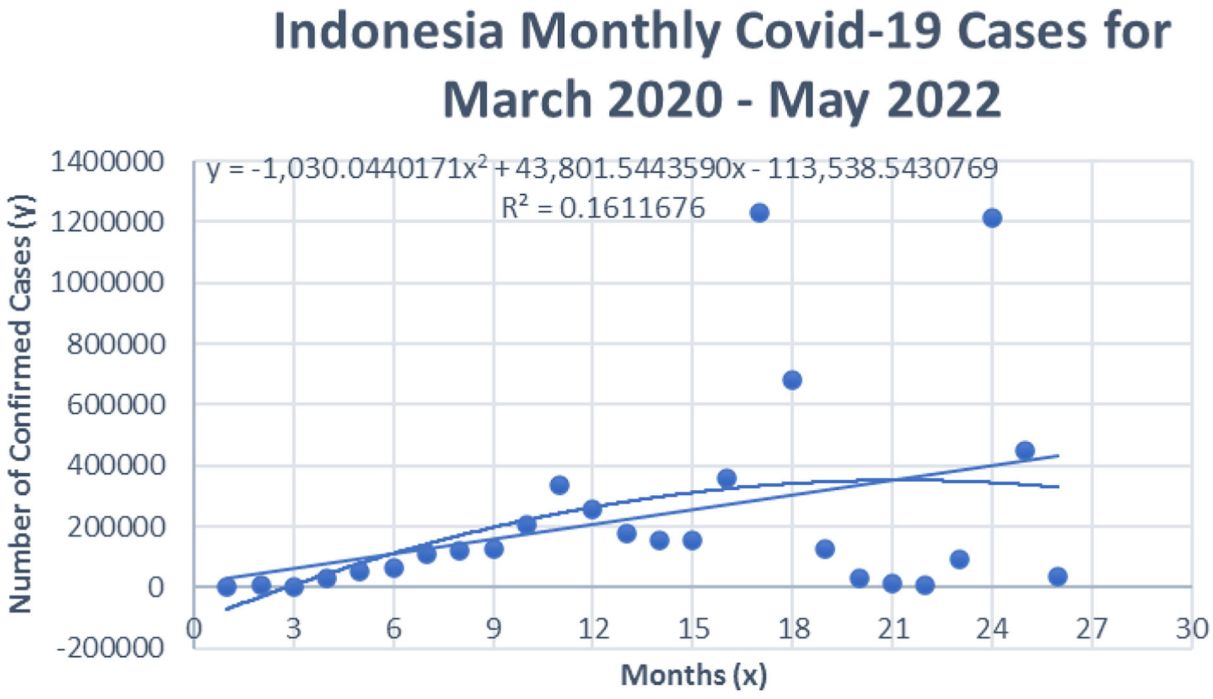


FIGURE 4
Nonlinear quadratic trend line for indonesia COVID-19 cases.

4.1. Trend line method

In finance and related areas, one of the easiest processes to boost the likelihood of making a successful trade is to understand the direction of an underlying trend because it assures that the overall market dynamics are in your favor. Trend lines are bounding lines that traders use to connect a sequence of prices of security on charts. It is created when three or more price pivot points or more can be connected diagonally. In this section, the proposed FMSD and existing least squares methods were employed to estimate data from Table 3. Microsoft Excel software was used to plot the trend line for data for the first 26 months. The graph demonstrated in Figure 4 was obtained by plotting the real data from Table 3 with x and y denoting the x-axis and y-axis respectively.

The efficiency of the proposed method is further demonstrated by comparing the approximation functions of FMSD with those of the trend line and least squares methods. Table 5 presents the estimation Point and relative Errors for the three methods based on the reserved data for the 27th month.

From Table 5, we can see that the error ε has been minimized which agrees with the main purpose of regression analysis. This shows that the proposed FMSD method is efficient and promising, and thus, can find a wider range of other real-life applications.

TABLE 5 Relative error analysis using the data of the 27th month.

Models	Sum of error	Average error
Least square	−195.0314769076	−7.50119802392
FMSD	−195.0314760235	−7.50119626000

5. Conclusions

In this paper, we have presented a spectral conjugate gradient method for solving unconstrained optimization problems by modifying the spectral parameter of the JYJLL method in Jian et al. [7]. Based on some conditions, the global convergence properties were established under a weak Wolfe line search. A numerical comparison of the proposed method with the JYJLL method shows that the proposed method is efficient, fast, and robust. Moreover, our proposed method can solve the COVID-19 case model in Indonesia.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

MM and FN: conceptualization. MM and IS: methodology, numerical experiments, and writing—original draft preparation. MM and DA: formal analysis. IS: application. All authors have read and agreed to the published version of the manuscript.

Funding

This research is funded by Hibah Riset Penugasan FMIPA UI (Grant No. 002/UN2.F3.D/PPM.00.02 /2022).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Disease (COVID-19) C (2022). Available online at; https://www.who.int/health-topics/coronavirus (accessed June 7, 2022).

2. Data CCP (2022). Available online at: https://www.worldometers.info/coronavirus/#countries (accessed June 7, 2022).

3. Aggarwal V, Arora G, Emadifar H, Hamasalh FK, Khademi M. Numerical simulation to predict COVID-19 cases in punjab. *Comput Math Methods Med.* (2022) 2022:7546393. doi: 10.1155/2022/7546393

4. Amar LA, Taha AA, Mohamed MY. Prediction of the final size for COVID-19 epidemic using machine learning: a case study of Egypt. *Infect Dis Model.* (2020) 5:622–34. doi: 10.1016/j.idm.2020.08.008

5. Sulaiman IM, Malik M, Awwal AM, Kumam P, Mamat M, Al-Ahmad S. On three-term conjugate gradient method for optimization problems with applications on COVID-19 model and robotic motion control. *Adv Continuous Discrete Models.* (2022) 2022:1–22. doi: 10.1186/s13662-021-03638-9

6. Andrei N. *Nonlinear Conjugate Gradient Methods for Unconstrained Optimization.* Berlin; Heidelberg: Springer (2020).

7. Jian J, Yang L, Jiang X, Liu P, Liu M. A spectral conjugate gradient method with descent property. *Mathematics.* (2020) 8:280. doi: 10.3390/math8020280

8. Zheng X, Shi J. A modified sufficient descent Polak–Ribiére–Polyak type conjugate gradient method for unconstrained optimization problems. *Algorithms.* (2018) 11:133. doi: 10.3390/a11090133

9. Polak E, Ribiere G. Note sur la convergence de méthodes de directions conjuguées. *Revue française d'informatique et de recherché opérationnelle Série rouge.* (1969) 3:35–43. doi: 10.1051/m2an/196903R100351

10. Polyak BT. The conjugate gradient method in extremal problems. *USSR Comput Math Math Phys.* (1969) 9:94–112. doi: 10.1016/0041-5553(69)90035-4

11. Hager WW, Zhang H. A survey of nonlinear conjugate gradient methods. *Pacific J Optim.* (2006) 2:35–58.

12. Malik M, Mamat M, Abas SS, Sulaiman IM, Sukono F. A new coefficient of the conjugate gradient method with the sufficient descent condition and global convergence properties. *Eng Lett.* (2020) 28:704–14.

13. Malik M, Sulaiman IM, Mamat M, Abas SS, Sukono F. A new class of nonlinear conjugate gradient method for unconstrained optimization and its application in portfolio selection. *Nonlinear Funct Anal Appl.* (2021) 26:811–37. doi: 10.22771/nfaa.2021.26.04.10

14. Abubakar AB, Kumam P, Malik M, Chaipunya P, Ibrahim AH. A hybrid FR-DY conjugate gradient algorithm for unconstrained optimization with application in portfolio selection. *AIMS Math.* (2021) 6:6506–27. doi: 10.3934/math.2021383

15. Abubakar AB, Kumam P, Malik M, Ibrahim AH. A hybrid conjugate gradient based approach for solving unconstrained optimization and motion control problems. *Math Comput Simulat.* (2021) 201:640–57. doi: 10.1016/j.matcom.2021.05.038

16. Abubakar AB, Malik M, Kumam P, Mohammad H, Sun M, Ibrahim AH, et al. A Liu-Storey-type conjugate gradient method for unconstrained minimization problem with application in motion control. *J King Saud Univer Sci.* (2022) 34:101923. doi: 10.1016/j.jksus.2022.101923

17. Malik M, Mamat M, Abas SS, Sulaiman IM, Sukono F. A new spectral conjugate gradient method with descent condition and global convergence property for unconstrained optimization. *J Math Comput Sci.* (2020) 10:2053–69.

18. Petrović MJ, Stanimirović PS. Accelerated double direction method for solving unconstrained optimization problems. *Math Problems Eng.* (2014) 2014:965104. doi: 10.1155/2014/965104

19. Petrović MJ. An accelerated double step size model in unconstrained optimization. *Appl Math Comput.* (2015) 250:309–19. doi: 10.1016/j.amc.2014.10.104

20. Petrović M, Ivanović M, Đorđević M. Comparative performance analysis of some accelerated and hybrid accelerated gradient models. *Univers Thought Publicat Natural Sci.* (2019) 9:57–61. doi: 10.5937/univtho9-18174

21. Petrović MJ. Hybridization rule applied on accelerated double step size optimization scheme. *Filomat.* (2019) 33:655–65. doi: 10.2298/FIL1903655P

22. Petrović MJ, Valjarević D, Ilić D, Valjarević A, Mladenović J. An improved modification of accelerated double direction and double step-size optimization schemes. *Mathematics.* (2022) 10:259. doi: 10.3390/math10020259

23. Mirhoseini N, Babaie-Kafaki S, Aminifard Z. A nonmonotone scaled fletcher-reeves conjugate gradient method with application in image reconstruction. *Bull Malays Math Sci Soc.* (2022) 45. doi: 10.1007/s40840-022-01303-2

24. Babaie-Kafaki S, Mirhoseini N, Aminifard Z. A descent extension of a modified Polak–Ribière–Polyak method with application in image restoration problem. *Optim Lett.* (2022) 2022. doi: 10.1007/s11590-022-01878-6

25. Jiang X, Liao W, Yin J, Jian J. A new family of hybrid three-term conjugate gradient methods with applications in image restoration. *Num Algorith.* (2022) 91. doi: 10.1007/s11075-022-01258-2

26. Ebrahimnejad A, Aminifard Z, Babaie-Kafaki S. A scaled descent modification of the Hestense-Stiefel conjugate gradient method with application to compressed sensing. *J New Res Math.* (2022). doi: 10.30495/jnrm.2022.65570.2211

27. Aminifard Z, Babaie-Kafaki S. Dai-Liao extensions of a descent hybrid nonlinear conjugate gradient method with application in signal processing. *Num Algorith.* (2022) 89:1369–87. doi: 10.1007/s11075-021-01157-y

28. Sulaiman IM, Malik M, Giyarti W, Mamat M, Ibrahim MAH, Ahmad MZ. The application of conjugate gradient method to motion control of robotic manipulators. In: *Enabling Industry 4.0 Through Advances in Mechatronics.* Singapore: Springer (2022). p. 435–45.

29. Awwal AM, Sulaiman IM, Malik M, Mamat M, Kumam P, Sitthithakerngkiet K. A spectral rmil+ conjugate gradient method for unconstrained optimization with applications in portfolio selection and motion control. *IEEE Access.* (2021) 9:75398–414. doi: 10.1109/ACCESS.2021.3081570

30. Deepho J, Abubakar AB, Malik M, Argyros IK. Solving unconstrained optimization problems via hybrid CD-DY conjugate gradient

methods with applications. *J Comput Appl Math*. (2022) 405:113823. doi: 10.1016/j.cam.2021.113823

31. Malik M, Sulaiman IM, Abubakar AB, Ardaneswari G, Sukono F. A new family of hybrid three-term conjugate gradient method for unconstrained optimization with application to image restoration and portfolio selection. *AIMS Math*. (2022) 8:1–28. doi: 10.3934/math.2023001

32. Sulaiman IM, Bakar NA, Mamat M, Hassan BA, Malik M, Ahmed AM. A new hybrid conjugate gradient algorithm for optimization models and its application to regression analysis. *Indon J Electr Eng Comput Sci*. (2021) 23:1100–9. doi: 10.11591/ijeecs.v23.i2.pp1100-1109

33. Hestenes MR, Stiefel E. Methods of conjugate gradients for solving linear systems. *J Res Natl Bureau Standards*. (1952) 49:409–36. doi: 10.6028/jres.049.044

34. Fletcher R, Reeves CM. Function minimization by conjugate gradients. *Comput J*. (1964) 7:149–54. doi: 10.1093/comjnl/7.2.149

35. Fletcher R. *Practical Methods of Optimization*. Chichester: John Wiley & Sons (2013).

36. Dai YH, Yuan YX. A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J Optim*. (1999) 10:177–82. doi: 10.1137/S1052623497318992

37. Liu Y, Storey C. Efficient generalized conjugate gradient algorithms, part 1: theory. *J Optim Theory Appl*. (1991) 69:129–37. doi: 10.1007/BF00940464

38. Rivaie M, Mamat M, June LW, Mohd I. A new class of nonlinear conjugate gradient coefficients with global convergence properties. *Appl Math Comput*. (2012) 218:11323–32. doi: 10.1016/j.amc.2012.05.030

39. Hager W, Zhang H. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM J Optim*. (2005) 16:170–92. doi: 10.1137/030601880

40. Kou CX, Dai YH. A modified self-scaling memoryless Broyden-Fletcher-Goldfarb-Shanno method for unconstrained optimization. *J Optim Theory Appl*. (2015) 165:209–24. doi: 10.1007/s10957-014-0528-4

41. Andrei N. New accelerated conjugate gradient algorithms as a modification of Dai-Yuan's computational scheme for unconstrained optimization. *J Comput Appl Math*. (2010) 234:3397–410. doi: 10.1016/j.cam.2010.05.002

42. Liu JK, Feng YM, Zou LM. A spectral conjugate gradient method for solving large-scale unconstrained optimization. *Comput Math Appl*. (2019) 77:731–9. doi: 10.1016/j.camwa.2018.10.002

43. Zoutendijk G. Nonlinear programming, computational methods. In: *Integer and Nonlinear Programming*. Amsterdam (1970). p. 37–86.

44. Andrei N. An unconstrained optimization test functions collection. *Adv Model Optim*. (2008) 10:147–61.

45. Moré JJ, Garbow BS, Hillstrom KE. Testing unconstrained optimization software. *ACM Trans Math Softw*. (1981) 7:17–41. doi: 10.1145/355934.355936

46. Dolan ED, Moré JJ. Benchmarking optimization software with performance profiles. *Math Program*. (2002) 91:201–13. doi: 10.1007/s101070100263

47. COVID-19 pandemic in Asia W (2022). Available online at: https://en.wikipedia.org/w/index.php?title=COVID19pandemicinAsia&oldid=1089905138 (accessed June 21, 2022).

48. by Country | Asia CCC (2022). Available online at: https://tradingeconomics.com/country-list/coronavirus-cases?continent=asia (accessed June 21, 2022).

49. Aisyah DN, Mayadewi CA, Diva H, Kozlakidis Z, Siswanto, Adisasmito W. A spatial-temporal description of the SARS-CoV-2 infections in Indonesia during the first six months of outbreak. *PLoS ONE*. (2020) 15:e0243703. doi: 10.1371/journal.pone.0243703

50. to COVID-19 in Indonesia (As of 4 April 2022) Indonesia ReliefWeb SUR (2022). Available online at: https://reliefweb.int/report/indonesia/situation-update-response-covid-19-indonesia-4-april-2022 (accessed June 7, 2022).

51. Data IWCDCDWV (2022). Available online at: https://covid19.who.int (accessed June 21, 2022).

52. Sulaiman IM, Mamat M. A new conjugate gradient method with descent properties and its application to regression analysis. *J Num Anal Ind Appl Math*. (2020) 12:25–39.

53. Worldometer ICCS (2022). Available online at: https://www.worldometers.info/coronavirus/country/indonesia/ (accessed June 22, 2022).

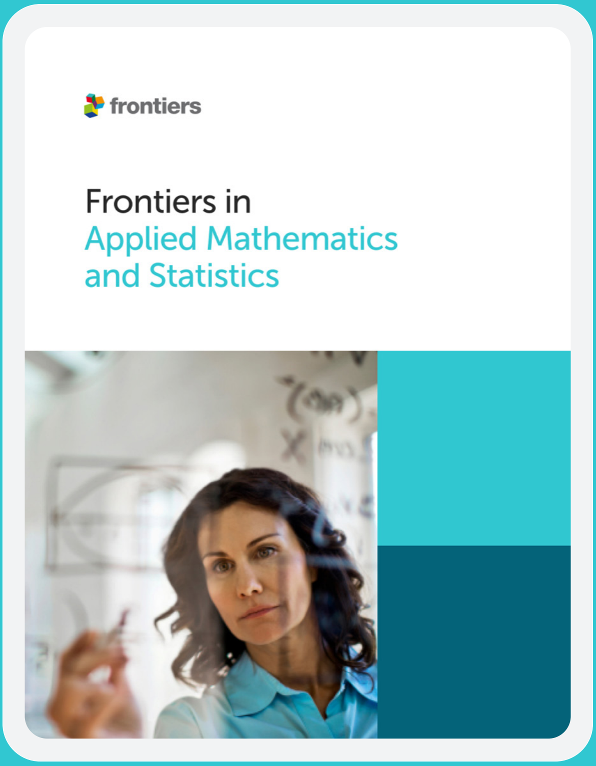
Frontiers in
Applied Mathematics and Statistics

Investigates both applied and applicable mathematics and statistical techniques

Explores how the application of mathematics and statistics can drive scientific developments across data science, engineering, finance, physics, biology, ecology, business, medicine, and beyond

Discover the latest Research Topics

See more →



frontiers

Frontiers in
Applied Mathematics
and Statistics

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact