TYPE Original Research
PUBLISHED 08 November 2022
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

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

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 $\mathbf{z}_{k-1}^T\mathbf{q}_{k-1} \le \mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|$ and $\mu > 0$, then

$$\begin{aligned}\mathbf{z}_{k-1}^T\mathbf{q}_{k-1} &= \mathbf{z}_{k-1}^T(\mathbf{h}_k - \mathbf{h}_{k-1}) \\ &= \mathbf{h}_k^T\mathbf{z}_{k-1} - \mathbf{h}_{k-1}^T\mathbf{z}_{k-1} \le \mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|,\end{aligned}$$

it implies

$$\mathbf{h}_k^T\mathbf{z}_{k-1} \le \mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\| + \mathbf{h}_{k-1}^T\mathbf{z}_{k-1}. \tag{10}$$

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

$$\begin{aligned}&\mathbf{h}_k^T\mathbf{z}_k \\ &= \mathbf{h}_k^T(-\theta_k^{FMSD}\mathbf{h}_k + \beta_k^{FMSD}\mathbf{z}_{k-1}) \\ &= -\theta_k^{FMSD}\|\mathbf{h}_k\|^2 + \beta_k^{FMSD}\mathbf{h}_k^T\mathbf{z}_{k-1} \\ &= -\left[1 - \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\right]\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \frac{(\mathbf{h}_k^T\mathbf{z}_{k-1})^2}{\|\mathbf{z}_{k-1}\|^2}}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\mathbf{h}_k^T\mathbf{z}_{k-1} \\ &= -\|\mathbf{h}_k\|^2 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\mathbf{h}_k^T\mathbf{z}_{k-1} \\ &\le -\|\mathbf{h}_k\|^2 + \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|} \\ &\times(\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\| + \mathbf{h}_{k-1}^T\mathbf{z}_{k-1}) \\ &= \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2(1-\cos^2\theta_k)}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|}\mathbf{h}_{k-1}^T\mathbf{z}_{k-1} \\ &-\|\mathbf{h}_k\|^2\cos^2\theta_k \le \frac{|\mathbf{h}_k^T\mathbf{z}_{k-1}|}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}\|\mathbf{h}_k\|^2 + \frac{\|\mathbf{h}_k\|^2\sin^2\theta_k}{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|} \\ &\mathbf{h}_{k-1}^T\mathbf{z}_{k-1} < 0.\end{aligned} \tag{12}$$

- **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} \\ &= -\|\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 \\ &\le \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 $k \ge 1$.

Next, we will prove the interval of β_k^{FMSD}. From formulas (12 and (13, we obtain the relation $\mathbf{h}_k^T\mathbf{z}_k \le \beta_k^{FMSD}\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}$. Furthermore, since $\mathbf{h}_k^T\mathbf{z}_k < 0$, we have $\beta_k^{FMSD} \le \frac{\mathbf{h}_k^T\mathbf{z}_k}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}$.

Now, from formulas (8) and (7), we get

$$\begin{aligned}\beta_k^{FMSD} &= \frac{\|\mathbf{h}_k\|^2 - \frac{(\mathbf{h}_k^T\mathbf{z}_{k-1})^2}{\|\mathbf{z}_{k-1}\|^2}}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}} \\ &= \frac{\|\mathbf{h}_k\|^2 - \|\mathbf{h}_k\|^2\cos^2\theta_k}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}} \\ &= \frac{\|\mathbf{h}_k\|^2\sin^2\theta_k}{\max\{\mu\|\mathbf{z}_{k-1}\|\|\mathbf{q}_{k-1}\|, \mathbf{z}_{k-1}^T\mathbf{q}_{k-1}\}} \ge 0.\end{aligned}$$

Thus, $0 \le \beta_k^{FMSD} \le \frac{\mathbf{h}_k^T\mathbf{z}_k}{\mathbf{h}_{k-1}^T\mathbf{z}_{k-1}}$ 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 $\mathcal{B}: = \{\mathbf{r} \in \mathbb{R}^n : f(\mathbf{r}) \le f(\mathbf{r}_0)\}$ 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*

$$\|\mathbf{h}(\mathbf{r}) - \mathbf{h}(\mathbf{s})\| \le L\|\mathbf{r} - \mathbf{s}\|, \ \forall \mathbf{r}, \mathbf{s} \in \mathcal{L}.$$

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*

$$\liminf_{k\to\infty} \|\mathbf{h}_k\| = 0. \tag{19}$$

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.

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)

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 (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

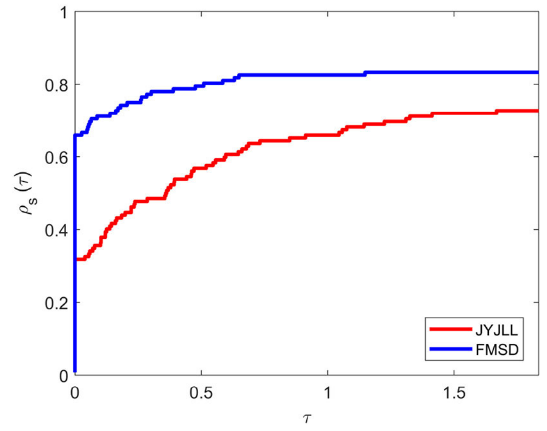

FIGURE 1
Performance profiles on NOI.

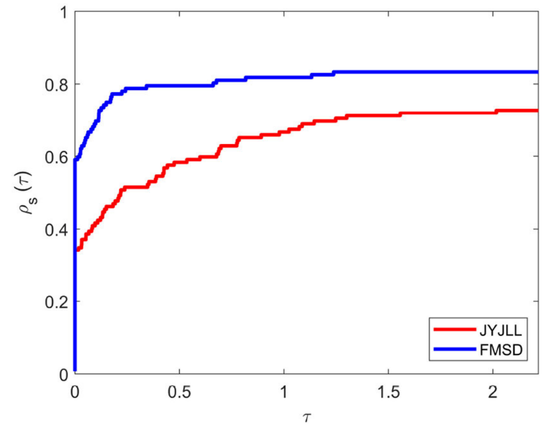

FIGURE 2
Performance profiles on NOF.

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.

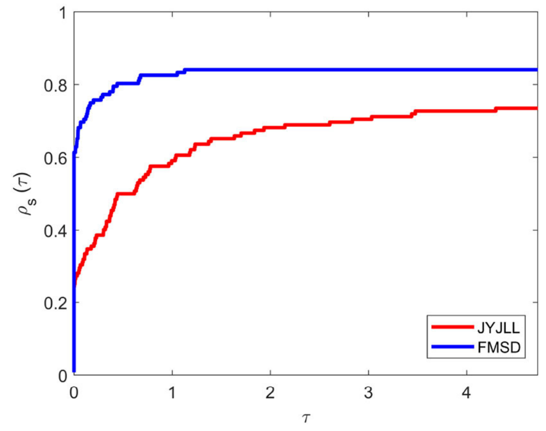

FIGURE 3
Performance profiles on CPU Time.

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.

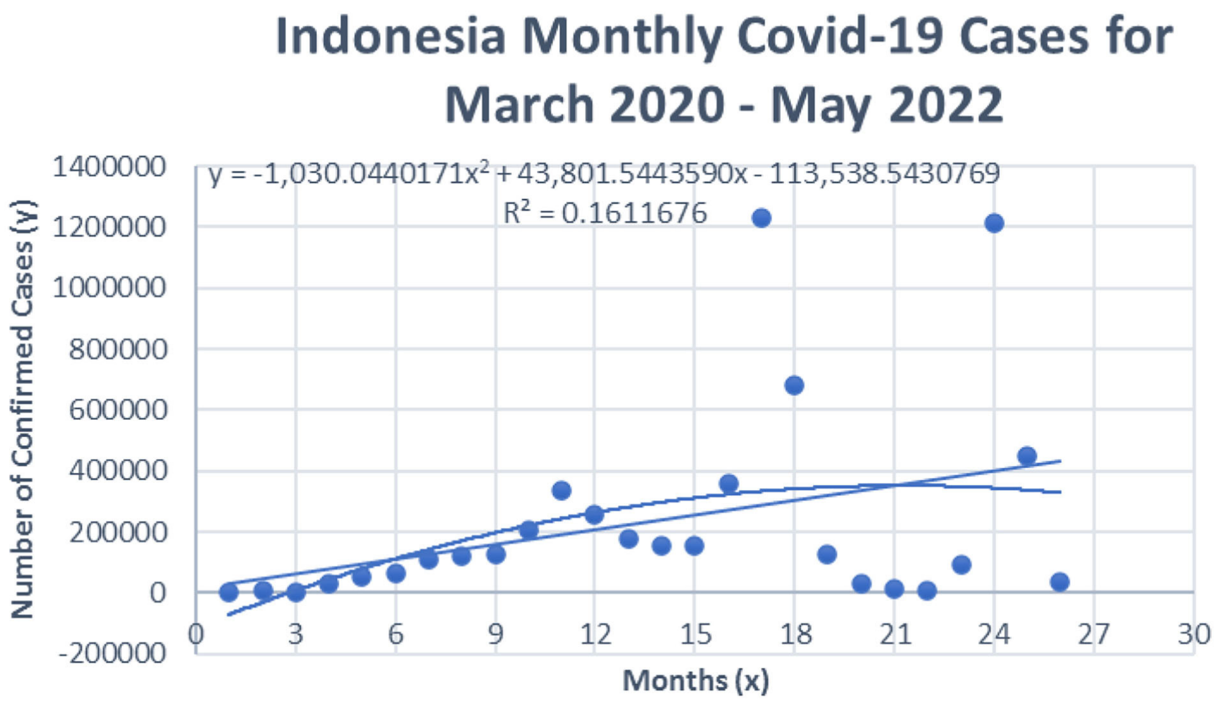

FIGURE 4
Nonlinear quadratic trend line for indonesia COVID-19 cases.

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).