



## OPEN ACCESS

## EDITED BY

Gernot R. Müller-Putz,  
Graz University of Technology, Austria

## REVIEWED BY

Syed Hammad Nazeer Gilani,  
Air University, Pakistan  
Miaoyun Zhao,  
Dalian University of Technology, China  
Wei Wang,  
Nanjing University of Aeronautics and  
Astronautics, China

## \*CORRESPONDENCE

Tianyu Liu  
✉ liuty@shmtu.edu.cn

RECEIVED 13 March 2024

ACCEPTED 08 May 2024

PUBLISHED 22 May 2024

## CITATION

Liu T, Wu Y, Ye A, Cao L and Cao Y (2024)  
Two-stage sparse multi-objective  
evolutionary algorithm for channel selection  
optimization in BCIs.  
*Front. Hum. Neurosci.* 18:1400077.  
doi: 10.3389/fnhum.2024.1400077

## COPYRIGHT

© 2024 Liu, Wu, Ye, Cao and Cao. This is an  
open-access article distributed under the  
terms of the [Creative Commons Attribution  
License \(CC BY\)](#). The use, distribution or  
reproduction in other forums is permitted,  
provided the original author(s) and the  
copyright owner(s) are credited and that the  
original publication in this journal is cited, in  
accordance with accepted academic practice.  
No use, distribution or reproduction is  
permitted which does not comply with these  
terms.

# Two-stage sparse multi-objective evolutionary algorithm for channel selection optimization in BCIs

Tianyu Liu<sup>1\*</sup>, Yu Wu<sup>1</sup>, An Ye<sup>1</sup>, Lei Cao<sup>1</sup> and Yongnian Cao<sup>2</sup>

<sup>1</sup>School of Information Engineering, Shanghai Maritime University, Shanghai, China, <sup>2</sup>TikTok Incorporation, San Jose, CA, United States

**Background:** Channel selection has become the pivotal issue affecting the widespread application of non-invasive brain-computer interface systems in the real world. However, constructing suitable multi-objective problem models alongside effective search strategies stands out as a critical factor that impacts the performance of multi-objective channel selection algorithms. This paper presents a two-stage sparse multi-objective evolutionary algorithm (TS-MOEA) to address channel selection problems in brain-computer interface systems.

**Methods:** In TS-MOEA, a two-stage framework, which consists of the early and late stages, is adopted to prevent the algorithm from stagnating. Furthermore, the two stages concentrate on different multi-objective problem models, thereby balancing convergence and population diversity in TS-MOEA. Inspired by the sparsity of the correlation matrix of channels, a sparse initialization operator, which uses a domain-knowledge-based score assignment strategy for decision variables, is introduced to generate the initial population. Moreover, a Score-based mutation operator is utilized to enhance the search efficiency of TS-MOEA.

**Results:** The performance of TS-MOEA and five other state-of-the-art multi-objective algorithms has been evaluated using a 62-channel EEG-based brain-computer interface system for fatigue detection tasks, and the results demonstrated the effectiveness of TS-MOEA.

**Conclusion:** The proposed two-stage framework can help TS-MOEA escape stagnation and facilitate a balance between diversity and convergence. Integrating the sparsity of the correlation matrix of channels and the problem-domain knowledge can effectively reduce the computational complexity of TS-MOEA while enhancing its optimization efficiency.

## KEYWORDS

multi-objective evolutionary algorithm, channel selection, two-stage framework, sparse initialization, score assignment strategy

## 1 Introduction

Brain-computer interface systems (BCIs) have garnered increasing attention within academic and industrial circles due to their broad real-world applications. By acquiring brain signals, BCIs facilitate external device control and communication without necessitating physical movement. For instance, individuals with paralysis can employ BCIs to manage external devices like wheelchairs, prosthetics, and robots, consequently enhancing their quality of life (Krishna Rao et al., 2022). Furthermore, BCIs find utility in monitoring patients' cerebral functions within the medical domain and delivering enhanced immersive experiences in the realm of gaming (Qu et al., 2023).

The majority of existing non-invasive BCIs utilize external sensors equipped with multiple channels (32 channels, 62 channels, or even more) for the acquisition of electroencephalography (EEG) signals (Sibilano et al., 2024). More channels lead to a more comprehensive capture of EEG signals. However, owing to the impact of the skull and scalp on electrical signal transmission, EEG signals obtained through external sensors frequently contain noise and extraneous information irrelevant to specific tasks. Moreover, the extensive number of EEG channels compounds the difficulties in data collection and significantly increases the computational complexity of processing this data, leading to increased and often unnecessary computational costs. Therefore, selecting appropriate channels (known as channel selection optimization) from the entirety has emerged as a pivotal challenge in the realm of BCIs (Almanza-Conejo et al., 2023).

A significant portion of research in channel selection optimization is grounded in EEG signal analysis (Martínez-Cagigal et al., 2022). The channel selection method based on signal analysis begins by extracting and selecting features from EEG signals, subsequently choosing the most suitable subset of channels for a specific task. On the one hand, such methods require users to have domain-specific expertise related to the task; otherwise, it may lead to selecting sub-optimal channel subsets. Furthermore, the pre-designed algorithmic workflow may become entirely inappropriate if the task changes. On the other hand, most signal analysis-based channel selection methods focus on a single optimization objective, with task accuracy often chosen as the optimization goal in many algorithms (Rocha-Herrera et al., 2022). However, in addition to task accuracy, the number of selected channels is also a crucial metric when conducting channel selection. This is because the number of channels adopted will determine the convenience of using BCI devices. However, there is typically a trade-off between the number of selected electrodes and task accuracy. Therefore, an efficient channel selection approach must strike a compromise between the number of selected channels and task accuracy since these two factors are mutually exclusive. In this context, the utilization of multi-objective evolutionary algorithms (MOEAs) (He et al., 2022), recognized for their efficiency in resolving problems with conflicting multiple objectives, has captured the attention of researchers. In most multi-objective channel selection algorithms, the number of selected channels (or the number of deleted channels) and the accuracy of tasks are directly employed to construct the multi-objective problem model (Alotaiby et al., 2015). However, researchers have found that utilizing the aforementioned multi-objective problem model can sometimes lead to premature convergence of MOEAs (Abdullah et al., 2022). Consequently, developing a well-designed and practical multi-objective problem model becomes a critical factor influencing the performance of multi-objective channel selection algorithms.

Studies have demonstrated that connectivity information can effectively capture the attributes of EEG signals, given that interactions and collaborations among various regions shape the neural activity in the brain (Moon et al., 2020). Recently, there has been a rising trend in utilizing the correlation matrix of channels to address channel selection optimization problems. This is primarily because the correlation matrix can depict the interactions and cooperative activities among different brain regions (Liu and Ye,

2023). It has been demonstrated that due to the non-uniform connectivity patterns in the brain, the correlation matrix of EEG signals is typically sparse (Liu et al., 2019). For example, Figure 1 exemplifies the correlation matrix of 62 EEG channels utilized in a fatigue detection task with a classification accuracy of 94%. In Figure 1, the red color represents that the corresponding two channels are entirely linearly correlated (linear correlation coefficient of 1), while the blue color indicates that the two channels are linearly independent (linear correlation coefficient of 0). It can be observed from Figure 1 that the majority of cells in the correlation matrix are depicted in blue, indicating that the correlation coefficients of the corresponding elements are close to zero, thereby demonstrating the sparse nature of the correlation matrix. However, few studies consider the sparsity of the correlation matrix when adopting it for solving channel selection problems.

This paper introduced a two-stage sparse multi-objective evolutionary algorithm (TS-MOEA), tailored for optimizing channel selection in BCIs. To prevent the algorithm from stagnating, TS-MOEA employs a two-stage framework. In this framework, the entire optimization process is divided into two phases, namely the early and late stages, with each stage addressing different multi-objective problem models. Specifically, in the early-stage phase, the adopted objective function is more sensitive to the deletion of channels to prevent the algorithm from falling into local optima. Furthermore, inspired by the sparsity of the correlation matrix of channels, a sparse initialization operator is employed when initializing the population in TS-MOEA. In the sparse initialization operator, a domain knowledge based strategy, which utilizes channels' positions and distance matrix, is used to assign scores to decision variables. Additionally, in the early stage of the algorithm, a *Score*-based mutation strategy is employed to enhance the search efficiency of the algorithm. In summary, the algorithm presented in this paper differs from existing multi-objective lead optimization algorithms in three main aspects. First, the proposed algorithm employs two distinct multi-objective optimization models, whereas current methods optimize for a single multi-objective model throughout the entire search process. Second, the proposed algorithm analyzes the sparsity of the channel correlation matrix and incorporates a sparsity-based strategy in the design of the operators to enhance the efficiency of the algorithm. Lastly, the proposed algorithm utilizes domain-specific knowledge to guide the search process of the algorithm. The primary contributions of this paper are outlined as follows:

1. A two-stage framework is employed in this study to assist the algorithm in escaping local optima. This framework divides the optimization process into the early and late stages, and different multi-objective problem models are used in the two stages.
2. Inspired by the sparsity observed in the correlation matrix of channels, a sparse initialization operator is introduced to create the initial population. Within this operator, a strategy based on domain knowledge is employed, leveraging channels' positions and distance matrix to allocate scores to decision variables.
3. A *Score*-based mutation strategy is employed to enhance the search efficiency in the early stage of TS-MOEA.
4. The performance of TS-MOEA and five other advanced multi-objective algorithms has been evaluated using a 62-channel EEG-based BCI system for a fatigue detection task.

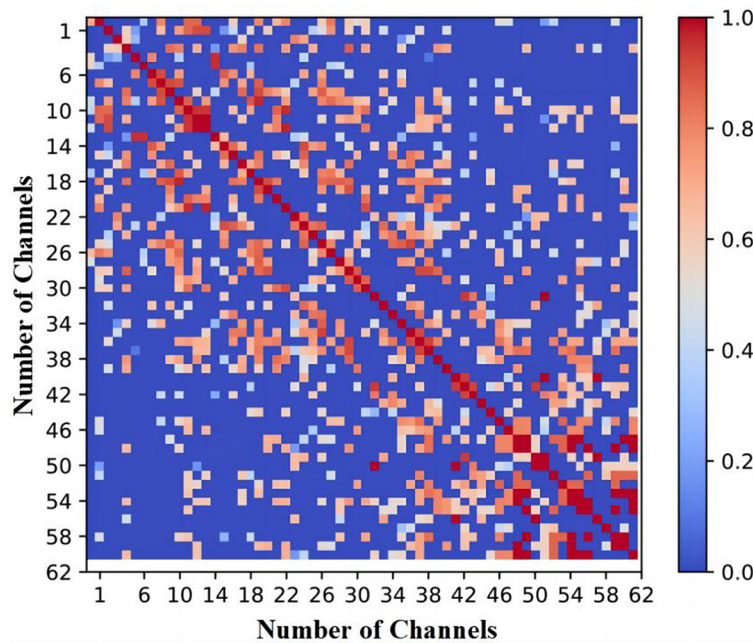


FIGURE 1  
Correlation matrix of 62 channels under 94% accuracy for a fatigue detection task.

The remainder of this paper is structured as follows: Section 2 presents the relevant background theory, followed by the description of the proposed TS-MOEA in Section 3. Section 4 covers the experiment and result analysis. Section 5 contains a discussion of the parameters and results, while Section 6 presents the concluding remarks.

## 2 Backgrounds

### 2.1 Acquisition and processing of EEG signals

In this study, EEG signals were collected using an ESI-64 high-resolution system (SynAmps2, Neuroscan) with 62 EEG channels (Chen et al., 2022). These 62 electrodes were positioned in accordance with the international 10–20 standard, as depicted in Figure 2. The initial sampling frequency was 1,000 Hz, which was down-sampled to 250Hz for data processing. Subsequently, the recorded signals underwent filtering with the frequency from 0 to 40 Hz. The raw EEG signals were sampled every 5 seconds, undergoing conversion from analog to digital signals through the utilization of a sampling window and a sliding window of 5 seconds. In this paper, the bilateral linked mastoid (LM) (Scannella et al., 2016), which is the average of the left and right mastoids, was used as the reference signal during the acquisition of EEG signals.

This study utilizes the correlation matrix to describe the characteristics of the collected EEG signals. In recent times, the Pearson Correlation Coefficient (PCC) (Pearson, 1895), along with the Phase Locking Value (PLV) (Aydore et al., 2013), and Transfer Entropy (TE) (Schreiber, 2000), have become extensively used

methods for calculating the correlation coefficient between EEG signals in BCIs.

PCC quantifies the linear correlation between two signals, with its value ranging from -1 to 1. A PCC value of 0 signifies that the signals are linearly uncorrelated. Conversely, PCC values of -1 and 1 indicate negative and positive linear relationships, respectively. Consider  $X_i = \{x_i^1, x_i^2, \dots, x_i^T\}$  as the EEG signal from the  $i^{\text{th}}$  channel, where  $T$  represents the signal's length, and  $\mu_i$  and  $\sigma_i$  are the mean and standard deviation of the  $i^{\text{th}}$  signal. The PCC value between  $X_i$  and  $X_k$  is computed using Equation (1).

$$PCC(i, k) = \frac{\frac{1}{T} \sum_{t=1}^T (X_i^t - \mu_i)(X_k^t - \mu_k)}{\sigma_i \sigma_k} \quad (1)$$

PLV is used to describe phase synchronization between two signals by averaging the absolute phase differences. PLV can be calculated using Equation (2). In this equation,  $\varphi_i^t$  denotes the phase of the signal at time point  $t$  for the  $i^{\text{th}}$  signal.

$$PLV(i, k) = \frac{1}{T} \left| \sum_{t=1}^T e^{j(\varphi_i^t - \varphi_k^t)} \right| \quad (2)$$

TE is a metric for quantifying the directed information flow from signal  $X_i$  to  $X_k$ , as delineated in Equation (3). Essentially, TE assesses the extent to which signal  $X_i$  can improve the prediction of signal  $X_k$ . A TE value of 0 indicates the absence of a causal relationship between the two time series, implying that knowing the past values of  $X_i$  may not aid in predicting  $X_k$ .

$$TE(i \rightarrow k) = \frac{1}{T-1} \sum_{t=1}^{T-1} p(X_i^t, X_k^t, X_k^{t+1}) \log \frac{p(X_k^{t+1} | X_k^t, X_i^t)}{p(X_k^{t+1} | X_k^t)} \quad (3)$$

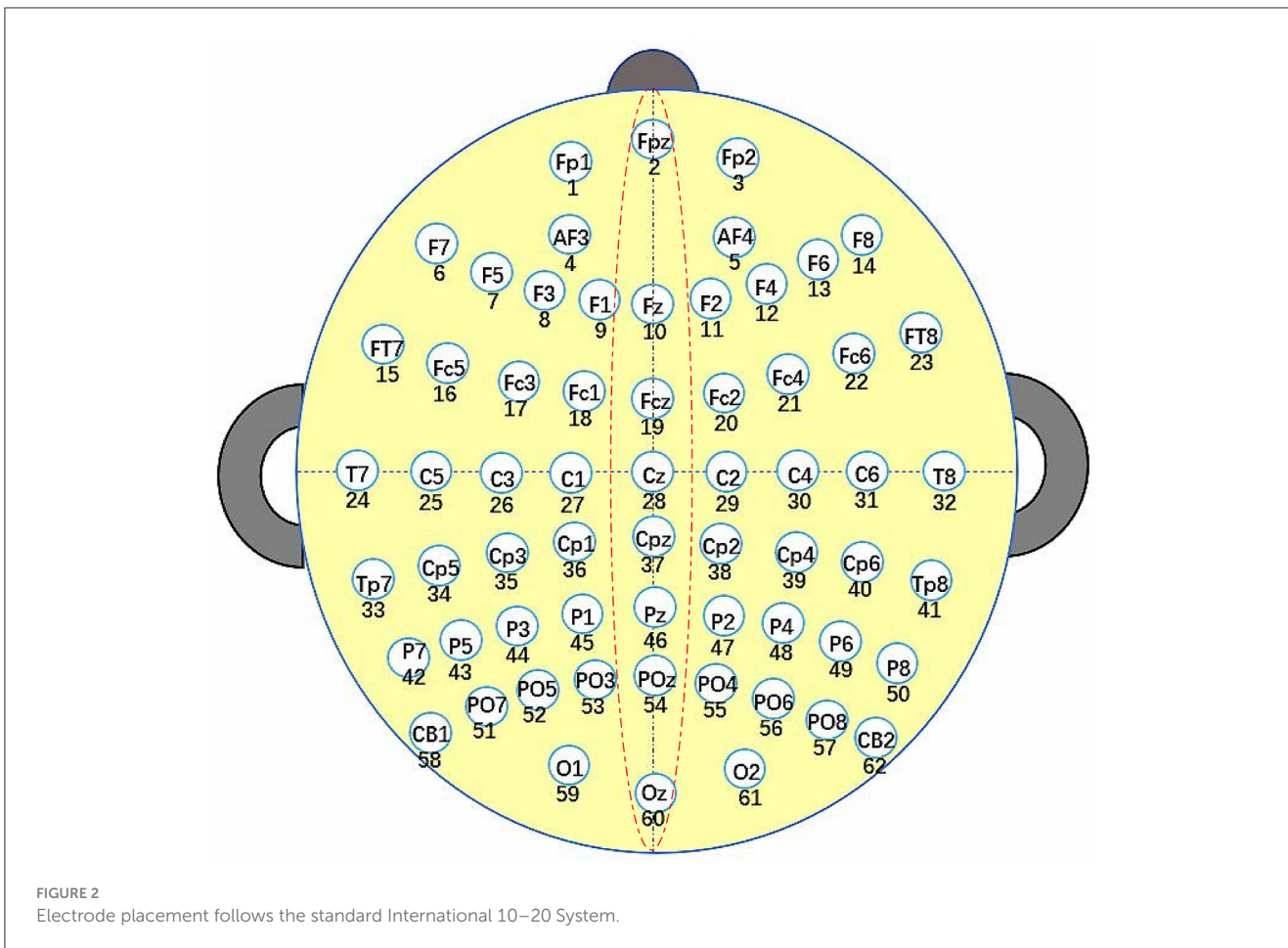


FIGURE 2 Electrode placement follows the standard International 10–20 System.

Researches have shown that the performance of TE is relatively worse than that of PCC and PLV. PLV performs better than PCC slightly (Gong et al., 2024). However, PCC is faster in terms of computational speed because of its simplicity (Maria et al., 2023). In this case, this paper utilizes PCC to obtain the correlation matrix of EEG signals in terms of both performance and computational speed.

no other solutions that can dominate  $x^*$ . Suppose a maximum MOP as shown in Equation (4),  $x$  dominates  $x^*$  if and only if  $\forall i \in \{1, 2, \dots, M\}, f_i(x) \geq f_i(x^*)$  and  $\exists i \in \{1, 2, \dots, M\}, f_i(x) > f_i(x^*)$ . Being population-based search methods, evolutionary algorithms (EAs) have proven to be efficient tools for tackling MOPs by generating a collection of candidate solutions within a single execution.

## 2.2 MOEAs

Multi-objective optimization problems (MOPs) are prevalent in various real-world scenarios, characterized by multiple conflicting objectives. The general formulation of a maximum MOP is represented in Equation (4), where  $x = (x_1, \dots, x_n) \in \Omega$  denotes the solution  $x$  within a search space of dimension  $n$ , and  $\Omega$  represents the feasible region in the search space.  $M$  corresponds to the number of objectives considered in the optimization problem. For an MOP with multiple exclusive objectives, the optimization algorithms can not find a single optimal solution that simultaneously optimizes all objectives. This is due to the exclusive between objectives, where enhancing the performance of one objective may lead to a decline in others. Consequently, the goal of MOEAs is to identify a set of Pareto optimal solutions.  $x^*$  is regarded as a Pareto optimal solution if there is

$$\text{Maximum } F(X) = (f_1(X), f_2(X), \dots, f_M(X)) \quad (4)$$

Current MOEAs can be categorized into three main types: dominance-based, decomposition-based, and index-based algorithms. For MOEAs belonging to the first category, the basic idea is to determine the priority of one solution by the dominant relation between the solution and the others. The typical algorithm in the first category is NSGA-II (Deb et al., 2002). In NSGA-II, a fast non-dominated sort method, which are widely adopted in dominance-based MOEAs, is proposed. Many improved algorithms have been submitted in recent years. For example, in CBGA-ES+ (Pradhan et al., 2021) proposes a hybrid selection strategy combining cluster-based methods and the traditional non-dominated elitist selection method to select parent solutions. In Premkumar et al. (2021), a MOSMA, which combines the Slime Mould Algorithm and the traditional NSGA-II, is proposed to solve MOPs in industries. In the CMMO (Ming et al., 2023)



algorithm, a cooperative evolution strategy, combined with customized environmental and mating selection, forms the basis for addressing MOPs. The algorithm utilizes dynamically adjusted relaxation factors to retain advantageous solutions with diverse decision spaces. This algorithm exhibits outstanding performance in solving multi-modal multi-objective problems. In ASDNSGA-II (Deng et al., 2022), a special congestion degree strategy and a new adaptive crossover operator are proposed to improve the performance of NSGA-II when handling multi-modal MOPs.

For MOEAs based on decomposition, the basic idea is to translate the original MOP into a set of single-objective problems [as seen approaches like MOEA/D (Zhang and Li, 2007)] or simple MOPs [as illustrated by MOEA/D-M2M and MOSOS/D (Liu et al., 2014; Ganesh et al., 2023)] with the help of weight vectors or reference points. Therefore, many improvements in this type of MOEAs focus on obtaining more appropriate weight vectors (reference points). For instance, Ma et al. (2020) propose an adaptive weight vector adjustment strategy, in which the weight vectors are periodically modified to enhance the searching capability of the algorithm. In MOEA/D-CSM (Liu et al., 2021), a dynamic reference points generation strategy, which considers the local knowledge in objective space, is proposed to obtain the reference points that can adapt well to MOPs with irregular Pareto fronts. In DMO-QPSO (You et al., 2021), a combination of the quantum-behaved particle swarm optimization (QPSO) algorithm and the MOEA based on decomposition (MOEA/D) is proposed. This integration aims to enable QPSO to effectively address MOPs while leveraging the strengths of QPSO. Additionally, the algorithm introduces some non-dominated solutions to guide other particles in the global best guidance group. The results indicate that the DMO-QPSO algorithm excels in addressing both two-objective and three-objective problems.

For index-based MOEAs, the additional indexes are adopted to determine the priority of solutions or guide the selection process in algorithms. Some representative indexes are hypervolume (HV) (While et al., 2006; Deist et al., 2023), inverted generation distance (IGD) (Zhou et al., 2006; Ishibuchi et al., 2019), dominance move (DoM) (Lopes et al., 2022), and R2 (Ma et al., 2018), and so on. In recent years, the hybrid index, which combines multiple indexes to improve search efficiency, has been proposed. For example, a hybrid index that combines HV and R2 has been adopted (Shang and Ishibuchi, 2020; Shang et al., 2020). Using HV to assess the distribution of the obtained Pareto fronts and R2 to measure the distance between these Pareto fronts and the ideal ones, the hybrid index facilitates algorithms in attaining a balance between convergence and population diversity.

## 2.3 Sparse MOEAs

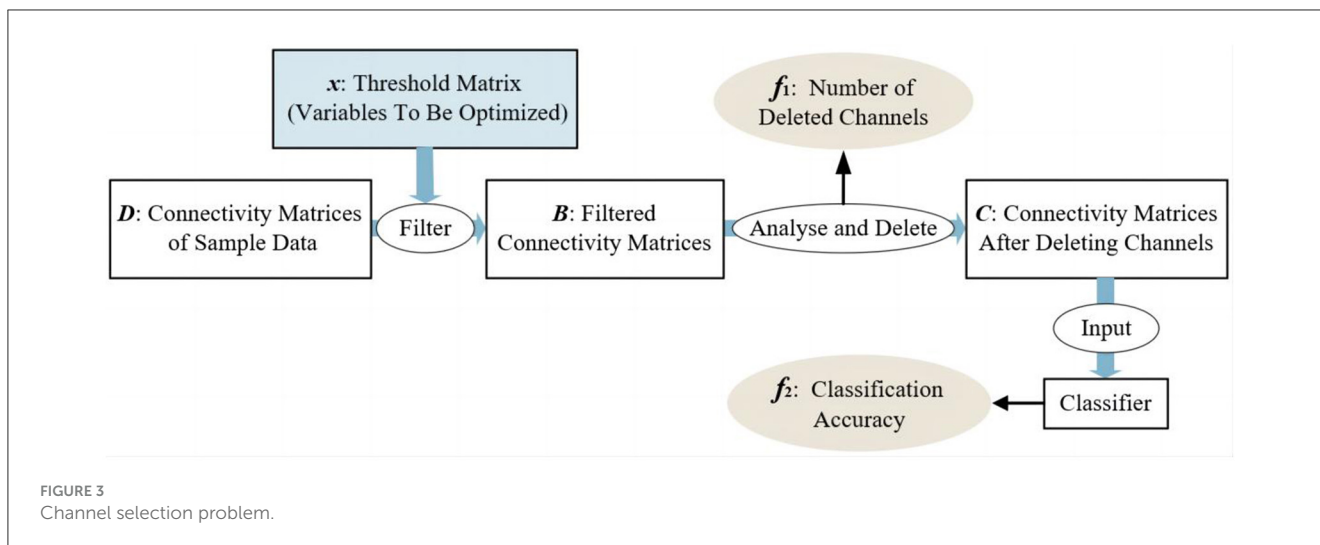
Studies have revealed that numerous MOPs possess sparse Pareto optimal solutions, particularly those with large-scale decision variables (Tian et al., 2021b). Such MOPs featuring sparsity are commonly referred to as sparse multi-objective optimization problems (SMOPs). In other words, most decision variables of the Pareto optimal solutions in SMOPs are 0. In this case, traditional MOEAs can not obtain satisfactory results when solving SMOPs.

This is because traditional MOEAs do not study the sparse distribution of Pareto optimal solutions and thus cannot effectively generate candidate solutions with sparsity in the evolution process.

In recent years, some variations of MOEAs have been applied to solving SMOPs successfully. These algorithms, called sparse multi-objective evolutionary algorithms (SMOEA), can be divided into two categories. In the first type, SMOEAs adopt the dimension reduction techniques that are commonly used in machine learning. For example, to reduce the number of sparse large-scale decision variables, MOEA/PSL (Tian et al., 2021a) leverages a denoising auto-encoder (DAE) followed by the utilization of a restricted Boltzmann machine (RBM) for acquiring insight into the sparse distribution of decision variables. PM-MOEA (Tian et al., 2022) adopts pattern mining techniques to identify the maximal and minimal candidate sets of non-zero decision variables from the population and apply specialized genetic operators to these patterns to achieve dimensional reduction. SMEA (Tian et al., 2023) proposes an effective approach for addressing sparse large-scale multi-objective evolutionary problems. The algorithm optimizes the binary vectors of each solution to estimate the sparse distribution of optimal solutions and introduces a rapid clustering method for significantly reducing the dimensionality of the search space. This algorithm partitions a substantial number of decision variables into multiple groups, where all variables within the same group are collectively represented by a single variable for optimization. This innovative strategy substantially diminishes the search space, thereby enhancing the convergence speed.

The search efficiency has been improving for the first type of SMOEAs since the dimension of search space has been reduced. However, some dimension reduction techniques may need high computational cost, and there is no sparsity-related knowledge as guidance information in the evolution process of the algorithms. In the second type, SMOEAs combines the conventional framework of MOEAs (such as NSGA-II) and a hybrid encoding method of solutions. For example, S-NSGA-II (Kropp et al., 2023) introduces a novel set of evolutionary operators, which include Varied Striped Sparse Population Sampling (VSSPS), Sparse Simulated Binary Crossover (S-SBX), and Sparse Polynomial Mutation (S-PM), to address SLMOPs. The aforementioned operators demonstrate remarkable efficacy in solving SLMOPs, particularly when evaluated using HV. In SparseEA, as introduced by Tian et al. (2020), a solution is represented by two components: a real vector for the original decision variables, and a binary vector, often referred to as a “mask vector,” which governs the solution’s sparsity. SparseEA2 adds a decision variable grouping strategy to accelerate the convergence speed of generating sparse Pareto optimal solutions. However, the decision variable grouping strategy in SparseEA2 is designed based on the random grouping method without considering the relation between variables. S-ECSO (Wang et al., 2022), an enhanced competitive swarm optimization approach, which adopts the strongly convex sparse operator (SCSparse), is designed to address SMOPs and exhibits outstanding performance.

As described in Section 1, the channel selection problems in BCIs is a typical SMOP. Based on the domain knowledge in the specific problem, this article proposes a two-stage sparse multi-objective optimization evolutionary algorithm, namely TS-MOEA. In TS-MOEA, both the sparsity and domain knowledge are



considered in the design of the fundamental operators. The detailed description of TS-MOEA is shown below.

## 3 Method

### 3.1 Formulation of two-objective channel selection optimization problem in two stages

This paper aims to select as few channels as possible with acceptable task accuracy. So, the number of deleted channels ( $f_1$ ) and the accuracy of tasks ( $f_2$ ) are the two maximized objectives that come to mind intuitively. Figure 3 illustrates the modeling process for the channel selection problem. Firstly, the raw signals are processed into sample data by computing the PCC values between each channel. Therefore, the sample data are all presented in the form of correlation matrices (as described in Section 2.1). Then, for the channel optimization problem, the threshold matrix  $x$  is considered as the decision variable that needs to be optimized. By filtering the sample data through the threshold matrix, it is easy to determine which channel can be deleted (Algorithm 1), and thus the value of  $f_1$  can be obtained, which is the number of deleted channels. Based on the channels that have been deleted, the subset of retained channels can be obtained. By using the data matrix of these selected/retained channels as the input for the classifier, the classification accuracy for a specific task can be achieved, denoted as  $f_2$ . In summary, the channel optimization problem is modeled as a maximization two-objective problem. As shown in Figure 3, the threshold matrix  $x$  contains the decision variables that need to be optimized, and  $x$  has the same size as the connectivity matrices of sample data. Where  $D = \{D_1, D_2, \dots, D_N\}$  and  $N$  is the number of samples,  $D_i (1 \leq i \leq N)$  is the correlation matrix for the  $i^{\text{th}}$  sample. After filtering  $D$  by  $x$ , one can obtain the set of the filtered correlation matrices, denoted as  $B$ , for all samples. Specifically,  $B = \{B_1, B_2, \dots, B_N\}$  and  $B_i (1 \leq i \leq N)$  is the filtered correlation matrix for the  $i^{\text{th}}$  sample. Then, the channels to be deleted can be determined by analyzing the filtered correlation matrices, thereby obtaining the number of

selected channels ( $f_1$ ). The detailed procedure of obtaining the number of deleted channels, i.e.,  $f_1$ , is given in Algorithm 1. As shown in Algorithm 1, if a channel is irrelevant to most channels, then this channel is most likely useless for the specific task and will be deleted (Lines 9, 10). In Line 10 of Algorithm 1, The value of  $s$  determines the difficulty level for channels to meet the deletion criteria. The correlation matrix after channel deletion  $C = \{C_1, C_2, \dots, C_N\}$  can be obtained based on  $B$ . For the  $k^{\text{th}}$  sample, if the  $j^{\text{th}}$  channel can be deleted, then  $C_k$  can be acquired by setting the elements in both the  $i^{\text{th}}$  row and the  $k^{\text{th}}$  column of  $B_k$  to 0 (Lines 17, 18). After that,  $C$  will be used as the input of classifiers, and then the accuracy of classification tasks ( $f_2$ ) can be obtained. The above-mentioned two-objective optimization problem can be formulated as shown in Equation (5). Please note that any classifier can be utilized for obtaining classification results. Since the focus of this paper does not center on the classifier itself, the classic support vector machine (SVM) is selected here. The hyperparameters used in SVM are obtained through the grid search method, in conjunction with 5-fold cross-validation. The hyperparameter determination process begins with establishing a range of potential values for each hyperparameter, forming a parameter grid. After evaluating each set of hyperparameter combinations by the 5-fold cross-validation method, the best values for the hyperparameters adopted in SVM will be obtained. The classification accuracy of SVM using the best hyperparameter will be regarded as  $f_2$ .

$$\begin{aligned} \text{Maximum } F(X) &= (f_1(X), f_2(X)) \\ f_1(X) &= m_d, m_d \text{ is the number of deleted channels} \\ f_2(X) &= \text{classifier}(C) \end{aligned} \quad (5)$$

$$f_1^*(X) = 0.5 * \frac{\text{zero}(C)}{N_C} + 0.5 * \frac{m_d}{m} \quad (6)$$

```

Require: Number of all channels, denoted as  $m$ ;
           Number of samples, represented by  $N$ ;
           Connection matrices of all samples,
           expressed as  $D = \{D_1, \dots, D_N\}$ ;
           Threshold matrix, represented as  $X$ ;
Ensure:  $f_1$ : Number of the deleted channels;
            $C = \{C_1, \dots, C_N\}$ : Connection matrices of all
           samples after channel deletion;
1: For  $k = 1:N$  % For each sample, obtain
   the filtered correlation matrix
2:    $B_k = D_k$ ;
3:   For  $i = 1:m$  % For each row in  $B_k$ 
4:     For  $j = 1:m$  % For each column in  $B_k$ 
5:       If  $B_k(i, j) < x(i, j)$ , then  $B_k(i, j) = 0$ ;
6:     End For
7:   End For
8:   For  $j = 1:m$  % For each row in  $B_k$ 
9:     Calculate the quantity of zero values in
     the  $j^{\text{th}}$  column and label this count as  $z_j$ .
10:    If  $z_j > s$  ( $s$  is the predefined threshold),
     then the  $j^{\text{th}}$  channel is considered to be
     optimized for the  $k^{\text{th}}$  sample;
11:    End For
12:  End For
13:   $m_d = 0$ ;
14:  For  $j = 1:m$  % For each channel
15:    If  $j^{\text{th}}$  channel is considered to be
     deleted for all samples, then the  $j^{\text{th}}$ 
     channel will be deleted and  $m_d = m_d + 1$ ;
16:    For  $k = 1:N$ 
17:       $C_k = B_k$ ;
18:      Set the elements in the  $j^{\text{th}}$  row and  $j^{\text{th}}$ 
     column of  $C_k$  to 0;
19:    End For
20:  End For
21:   $f_1 = m_d$ .

```

**Algorithm 1.** The detailed procedure of obtaining the number of deleted channels.

$$\begin{aligned}
 \text{Maximum } F(X) &= (f_1^*(X), f_2(X)) \\
 f_1^*(X) &= 0.5 * \frac{\text{zero}(C)}{N_C} + 0.5 * \frac{m_d}{m} \\
 f_2(X) &= \text{classifier}(C)
 \end{aligned} \quad (7)$$

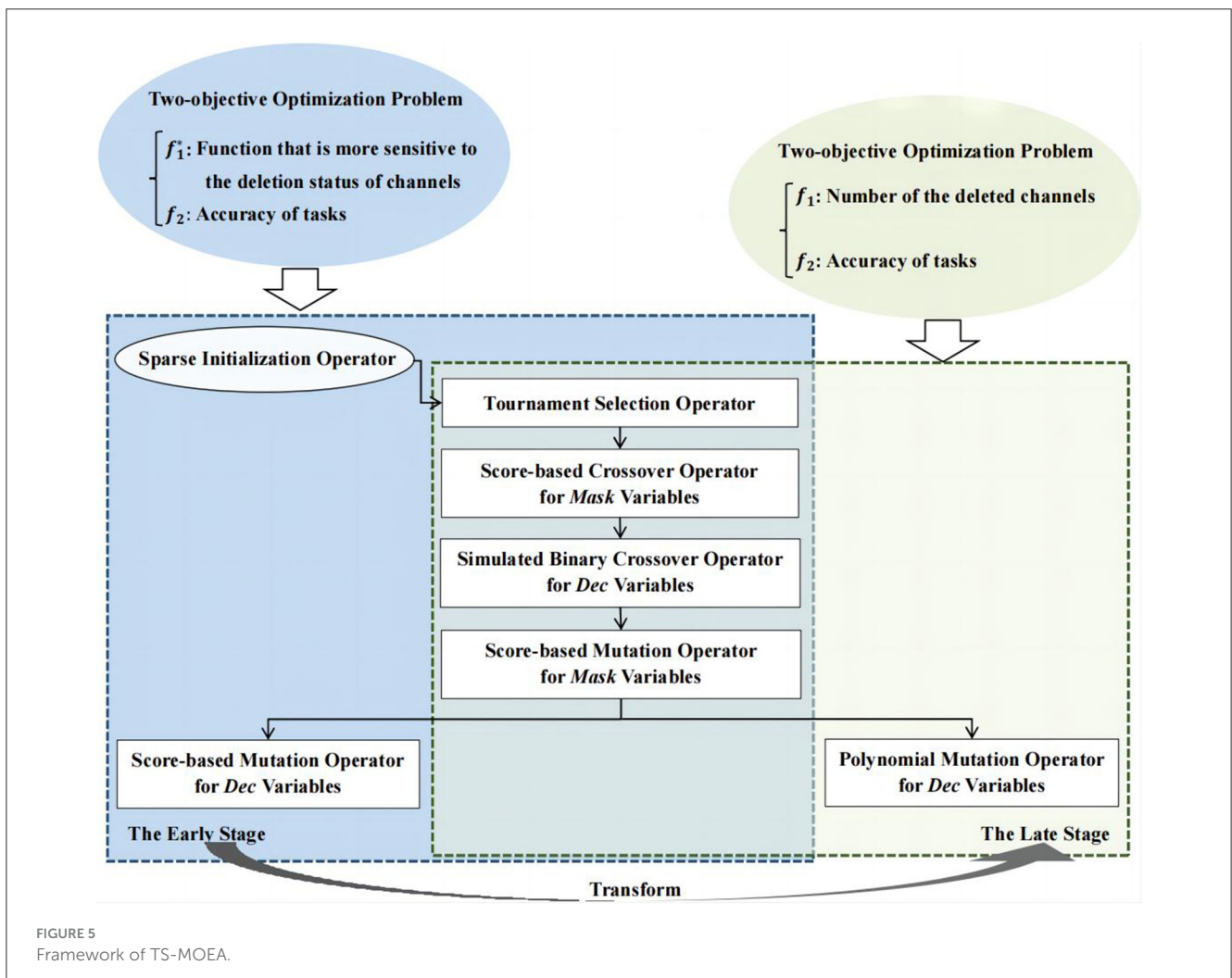
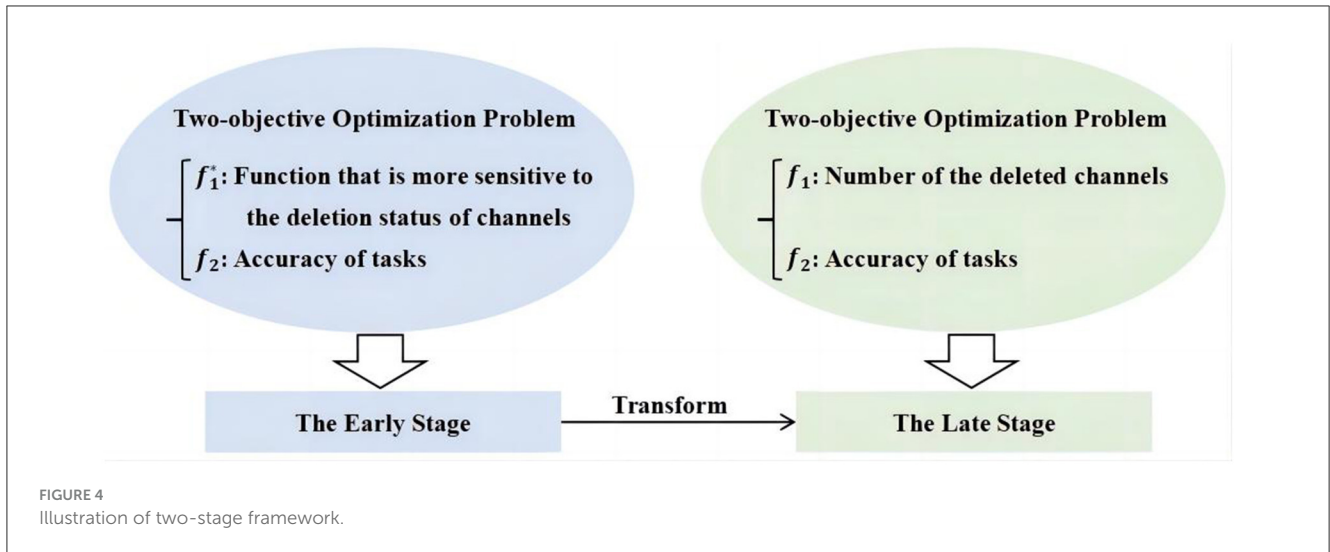
However, as shown in Algorithm 1, a channel can only be deleted if it is unanimously agreed upon by all samples. Particularly, when  $s$  is set to a large value, meeting the deletion criteria for channels becomes even more challenging. This difficulty results in MOEAs encountering stagnation when optimizing  $f_1$  (number of deleted channels). To address this problem, a novel objective function  $f_1^*$  is introduced, which offers higher sensitivity in reflecting the deletion status of channels. As expressed in Equation (6),  $\frac{\text{zero}(C)}{N_C}$  signifies the ratio of zero elements in  $C$ . Specifically,  $\text{zero}(C)$  represents the count of zero elements, and  $N_C$  is the total number of elements in  $C$ . Let  $m_d$  denote the number of deleted channels and  $m$  denote the total count of

channels. Then,  $\frac{m_d}{m}$  represents the proportion of deleted channels to the total channels. In this case, the multi-objective problem can be formulated as shown in Equation 7. In Equation 7, the number of deleted channels ( $f_1$ ) from the original Equation (5) is transformed into  $f_1^*$ , which represents the weighted average sum of the proportion of zero elements in  $C$  and the proportion of deleted channels to the total. After this transformation, the first objective function in the two-objective optimization model has shifted from a discrete integer search space to a continuous real number search space, which reduces the risk of the algorithm falling into a locally optimal solution. Therefore, compared to the two-objective problem model in Equation (5), the model in Equation (7) is more sensitive to the deletion status of channels, rendering it less susceptible to stagnation. Hence, this paper introduces a two-stage framework, as illustrated in Figure 4, employing different two-objective problem models in the early and late stages of the proposed algorithm.

### 3.2 Framework of TS-MOEA

In this paper, a two-stage sparse multi-objective evolutionary algorithm, named TS-MOEA, is introduced to address channel selection problems in BCIs. As illustrated in Figure 5, TS-MOEA adopts a two-stage framework comprising the early and late stages, each dedicated to distinct optimization problem models. It also can be observed from Figure 5 that the early and late stages share most operators. Specifically, in addition to the sparse initialization operator, the only difference between the two stages is the mutation of *Dec* variables. Furthermore, due to the sparsity of the correlation matrix, TS-MOEA adopted a hybrid representation of decision variables, which contains *Dec* variables (real numbers) and *Mask* variables (binary numbers). Algorithm 2 gives the detailed procedure of TS-MOEA.

In TS-MOEA, the output population obtained in the first stage becomes the input population for the late stage. To ensure population diversity in the late stage, TS-MOEA adopts a transformation condition between the two stages, which takes into consideration both the total of consumed function evaluations (*FE*) and (*NDC*), as shown in Line 4 of Algorithm 2. *NDC* represents the number of different  $f_1$  values obtained by *POP*. To ensure population diversity in the late stage, TS-MOEA adopts a transformation condition between the two stages, which takes into consideration both *FE* and *NDC*, as shown in Line 4 of Algorithm 2. Where *FE* is the current number of function evaluations consumed by the algorithm and *NDC* represents the number of different  $f_1$  values obtained by *POP*. Since  $f_1$  is one of the objective functions optimized by TS-MOEA in the late stage, a larger *NDC* implies that the population exhibits better diversity in the late stage of TS-MOEA. If the number of deleted channels is equal to  $m$ , i.e., all channels are removed, this is nonsensical. Moreover, since TS-MOEA is designed based on the correlation matrix between channels, this implicitly presupposes that the number of retained channels is greater than or equal to 2. Therefore, the possible values for the number of deleted channels can be any integer within the range  $[0, m - 2]$ . When the *NDC*



is  $m - 2$ , it indicates that the population generated in the first stage is sufficiently diverse to serve as the input population for the next stage. Moreover, if the number of the consumed function evaluations of the early stage exceeds the preset threshold, i.e.,

$\mu \times MaxFE$ , the algorithm can also transfer from the early stage to the late stage. In this case,  $\mu$  controls the transformation between the two stages, and its value has been investigated in detail in Section 4.2.



**Require:**  $N$  (representing the size of the population);  
 $G$  (indicating the arrangement of 62 electrodes following the international 10-20 standard);  
 $m$  (number of channels, i.e.,  $m = 62$  in this paper);  
**Ensure:** REP (the Pareto-optimal Set)

```

1: [POP,Score] = Sparse-Initialization( $N,G$ ); %
   Algorithm 3
2: Obtain REP as the non-dominated solution in POP;
3:  $NDC = \text{Unique}(f_1(\text{POP}))$  ;
4: While  $FE < \mu \times \text{MaxFE}$  or  $NDC < m - 2$  % The early
   stage
5:   Offspring =  $\emptyset$ ;
6:   Obtain  $2N$  parent individual from POP by
   binary tournament selection operator;
7:   For each pair of parent in individual(denoted
   as  $p$  and  $q$ ):
8:      $o.Dec \leftarrow \text{SBX\_Crossover}(p.Dec, q.Dec)$ ; % Crossover
   for  $Dec$  variables
9:      $o.mask \leftarrow p.mask$ ;
10:    if  $\text{rand}() < 0.5$  %
   Crossover for  $Mask$  variables
11:      Select two elements from the nonzero
   elements in
    $p.Mask \cap q.Mask$  randomly;
12:      Set the corresponding value in  $o.mask$ 
   to 0 for the selected element with a
   bigger  $Score$  value;
13:    Else
14:      Select two members from the nonzero
   elements in
    $q.Mask \cap \overline{p.Mask}$  randomly and set
   the values of the selected members in
    $o.mask$  to 1;
15:    End If
16:    If  $\text{rand}() < 0.5$  % Mutation for  $Mask$ 
   variables
17:      Randomly select two members from the
   nonzero elements in  $o.Mask$ ;
18:      Set the corresponding value in  $o.mask$ 
   to 0 for the selected element with a
   bigger  $Score$  value;
19:    Else
20:      Randomly select two members from the
   nonzero elements in  $o.Mask$ ;
21:      Set the corresponding value in  $o.mask$ 
   to 1 for the selected element with a
   smaller  $Score$  value;
22:    End If
23:    Divide  $o.Dec$  into different groups by
   ordered grouping and select one group
   randomly, then store the selected  $Dec$ 
   variables in  $SelectList$ ;
   % Mutation for  $Dec$  variables
   (Algorithm 4)

```

```

24:      $o.Dec \leftarrow \text{Score\_Mutation}(o.dec, \text{Score}, \text{SelectList})$ ;
25:     Offspring  $\leftarrow$  Offspring  $\cup o$ ;
26:   End For
27:    $POP \leftarrow$  Select the best  $N$  individuals from  $POP$ 
   and Offspring;
28:    $REP \leftarrow$  Non-dominated solution set in  $POP$  and
    $REP$ ;
29:    $NDC = \text{Unique}(f_1(\text{POP}))$ ;
30: End While
31: While  $FE < \text{MaxFE}$  % The late stage
32:   Offspring =  $\emptyset$ ;
33:   For the selected parent individuals,
   perform the crossover operators for  $Dec$ 
   variables
   and perform the crossover and mutation
   operators for  $Mask$  variables according
   to
   Lines 8-22;
   % Mutation for  $Dec$  variables in the late
   stage
34:   Mutate the  $Dec$  variables of by the
   polynomial mutation operator;
35:   Obtain  $POP$  and  $REP$  by Lines 27-28.
36: End While

```

Algorithm 2. Procedure of TS-MOEA.

TS-MOEA introduces a sparse initialization operator to generate the initial population for channel selection problems (Line 1 in Algorithm 2). In the sparse initialization operator, each decision variable will be assigned a  $Score$  value, which is calculated according to the problem-domain knowledge. The detailed description of the sparse initialization operator is given in Section 3.3. Both the early and late stages in TS-MOEA adopt the binary tournament selection operator (Lavinias et al., 2018) to obtain parent individuals (Line 6). The crossover and mutation for  $Dec$  and  $Mask$  variables utilize different strategies. Specifically, the simulated binary crossover operator (Deb and Beyer, 2001; Zhassuzak et al., 2024) is adopted for  $Dec$  variables (Line 8), while the  $Score$ -based crossover operator, which is inspired by SparseEA2, is utilized for  $Mask$  variables (Lines 10–15). The mutation for  $Mask$  variables is implemented by the  $Score$ -based mutation operator as shown in Lines 16–22. To balance convergence and population diversity, a  $Score$ -based mutation and the conventional polynomial mutation operators are utilized for  $Dec$  variables in the early and late stages of TS-MOEA, respectively (Lines 24, 34). The description of the proposed  $Score$ -based mutation operator has been given in Section 3.4. TS-MOEA utilizes the sequential grouping strategy (Zille et al., 2016) to divide decision variables into groups. Specifically, if it is required to split  $d$  decision variables into  $k$  groups, then the first  $[d/k]$  decision variables will be classified into the first group, the next  $[d/k]$  decision variables will be classified into the second group, and so on. Here,  $[d/k]$  denotes the integer closest to  $d/k$ . Since the binary tournament selection, simulated binary crossover, and polynomial mutation operators are widely adopted in various MOEAs, their details will not be presented here to save space.

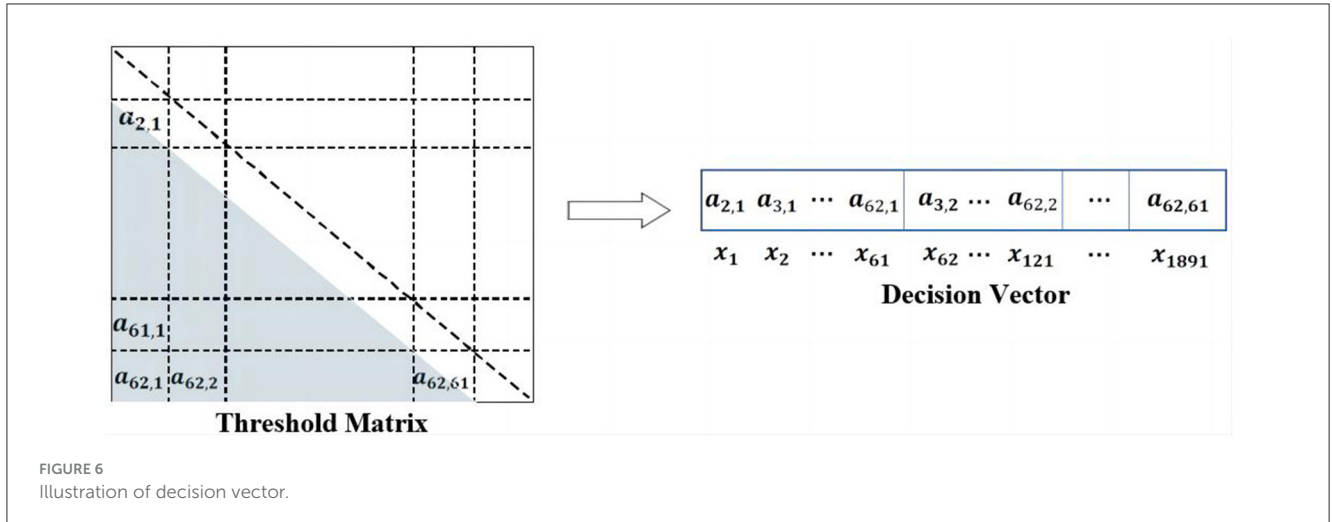


FIGURE 6  
Illustration of decision vector.

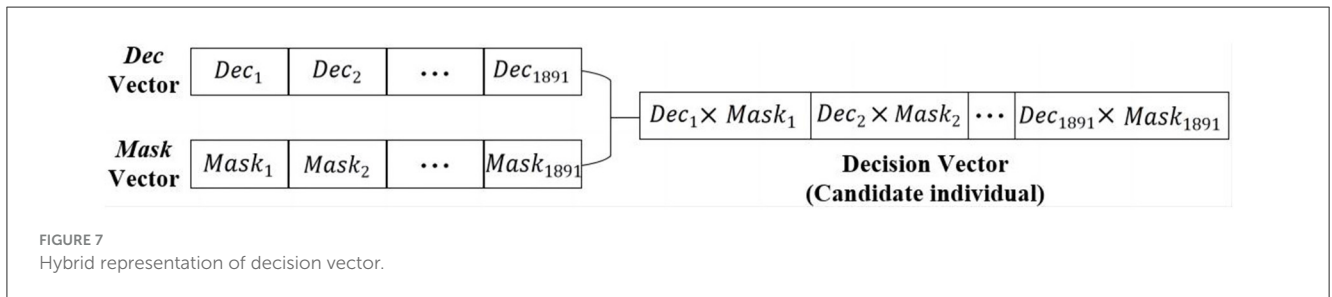


FIGURE 7  
Hybrid representation of decision vector.

### 3.3 Sparse initialization operator

In the proposed TS-MOEA, the threshold matrix  $x$  is the optimization target, as shown in Figure 3.  $x$  is employed to filter the correlation matrix of samples. Since the correlation matrix is symmetric,  $x$  will be rearranged as a decision vector. For instance, in this paper, the correlation matrix is  $62 \times 62$  due to the utilization of 62 channels. Therefore, the size of the decision vector will be  $1 \times 1,891$ , as shown in Figure 6. Inspired by SparseEA2, this paper adopted a hybrid representation of decision variables, which contains real variables (*Dec* vector) and binary variables (*Mask* vector). As illustrated in Figure 7, both the *Dec* vector and the *Mask* vector share the same size as the decision vector. The actual decision vector is obtained by multiplying corresponding elements from the *Dec* and *Mask* vectors.

Algorithm 3 provides a detailed procedure of the proposed sparse initialization operator. In this operator, the first step is to calculate the *Score* value of each variable in the decision vector (Lines 2–6). The *Score* values will later be used to determine whether elements in the *Mask* vector should be set to 0. Research has revealed that the relationship between brain regions relates to their location and length from each other (van den Broek et al., 1998; Reznik and Allen, 2018). Therefore, the calculation of *Score* values in this paper is based on domain-specific knowledge, which includes the location of channels and the distances between channels.

The calculation of *Score* values of the decision variables are given in Equations (8), (9). As presented in Equation (8),

$G_{Channel_k}$  represents the position of channel  $k$ , which can be acquired according to the international 10–20 standard.  $\|G_{Channel_k} - G_{Channel_l}\|_2$  signifies the Euclidean distance between  $G_{Channel_k}$  and  $G_{Channel_l}$ . In Equation (9),  $Max(DM)$  and  $Min(DM)$  denote the maximum and minimum distances between channels.  $Location_k = 1$  denotes that  $Channel_k$  is located in the left hemisphere, while  $Location_k = -1$  indicates that  $Channel_k$  is located in the right hemisphere, respectively. If  $Channel_k$  is positioned in the inter-hemispheric junction area of the brain, as illustrated by the dotted circles in Figure 2, then  $Location_k$  will be set to 0. As indicated in Equation (9), two channels located in different cerebral hemispheres have higher *Score* values compared to channels situated in the same hemisphere. Additionally, channels that are further apart have higher *Score* values. In Equation (9),  $R$  is the preset channel radius, whose value has been investigated in Section 4.3. For  $Channel_k$  and  $Channel_l$ , the larger the *Score* value, the easier it is for the corresponding element in *Mask* to be 0 (Lines 13–17) and the easier it is for the correlation coefficient of  $Channel_k$  and  $Channel_l$  to be 0 after filtering.

$$DM(k, l) = \|G_{Channel_k} - G_{Channel_l}\|_2 \quad (8)$$

$$Score(k, l) = \begin{cases} \frac{DM(k, l) - R}{2(Max(DM) + R)} & \text{if } Location_k = Location_l \\ \frac{DM(k, l)}{2(Max(DM) + R)} & \text{if } Location_k \neq Location_l \text{ and } DM(k, l) < R \\ \frac{DM(k, l) + R}{2(Max(DM) + R)} & \text{if } Location_k \neq Location_l \text{ and } DM(k, l) \geq R \end{cases} \quad (9)$$

```

Require:  $N$  (Population size);
            $D$  (Dimension of decision vectors);
            $m$  (Number of channels);
            $G$  (Position of channels according to
           the international 10-20 standard);
Ensure:  $POP$  (Initial population);
            $S$  (Score vector);
1: Calculate the value of distance matrix  $DM$ 
   according to  $G$  by Equation (8);
2: For  $k=1:m$ 
3:   For  $l=1:m$ 
4:     Calculate  $Score(k,l)$  by Equation (9);
5:   End For
6: End For
7: Convert the score matrix to score vector, denoted
   as  $S$ , as shown in Figure 6;
8: Generate  $N$   $Dec$  vectors, and each element in  $Dec$  is
   randomly generated within  $[0,1]$ ;
9: Generate  $N$   $Mask$  vectors, and each element in  $Dec$ 
   is set to 1;
10: For  $i=1$  to  $N$ 
11:   For  $j=1$  to  $rand \times D$ 
12:     Select two decision variables, namely  $m$ 
     and  $n$ , randomly;
13:     If  $S_m(j) \leq S_n(j)$  then
14:       Set the  $n^{th}$  element in the  $Mask$  vector of
     the  $i^{th}$  individual to 0;
15:     else
16:       Set the  $m^{th}$  element in the  $Mask$  vector
     of the  $i^{th}$  individual to 0;
17:     End If
18:   End For
19: End For
20: Obtain  $POP$ , where the  $i^{th}$  solution in  $POP$  is
   acquired by multiplying the  $i^{th}$   $Dec$  vector and the
    $i^{th}$   $Mask$  vector, as illustrated in Figure 7.

```

Algorithm 3. Sparse initialization operator.

### 3.4 Score-based mutation operator

As explained in Section 3.2, the output population from the early stage in TS-MOEA serves as the input population for the late stage. Hence, the quality of the population acquired in the early stage significantly impacts the ultimate performance of the proposed TS-MOEA. To effectively leverage the problem-domain knowledge to steer the search process of TS-MOEA, a *Score*-based mutation operator, which utilizes the *Score* values of decision variables, is introduced for *Dec* vectors in the early stage of TS-MOEA. Algorithm 4 presents the detailed procedure of the *Score*-based mutation operator.

For  $o.Dec$  (the *Dec* vector of individual  $o$ ), if  $sc_j$  is large, i.e., the corresponding two channels are close to each other, then  $o.Dec_j$  will have a greater probability to be a large value after mutation (Lines 4, 8 in Algorithm 4). In this case, the two channels that are related to  $Dec_j$  tend to be regarded as uncorrelated after filtering (Line 5 in Algorithm 4). In Algorithm 4,  $\alpha$  controls the magnitude of mutation and is set to 0.1 empirically.

```

Require:  $S = [sc_1, sc_2, \dots, sc_D]$  (Score vector);
            $o.Dec$  (Dec vector of individual  $o$ )
            $SelectList$  (Index of the elements that are
           selected to be mutated in  $o.Dec$ );
Ensure:  $o.Dec$ 
1: For  $j \in SelectList$ 
2:   Generate numbers  $r, r_1, r_2$ , each number is randomly
   selected within  $[0,1]$ ;
3:   If  $r < 0.5 + sc_j$ 
4:      $\Delta = \alpha \cdot r_1$ 
5:   Else
6:      $\Delta = -\alpha \cdot r_2$ 
7:   End If
8:    $o.Dec_j \leftarrow o.Dec_j + \Delta$ 
9: End For

```

Algorithm 4. *Score*-based mutation operator.

## 4 Results

### 4.1 EEG data and parameter settings

The unprocessed EEG signals were gathered from a group of 9 participants aged between 21 and 30 during a fatigue detection task. Throughout the data collection phase, participants underwent a wake-sleep-wake cycle post-lunch, a period when many people typically experience fatigue symptoms. For experiment integrity, all volunteers were required to wake up before 8:30 a.m. and abstain from alcohol and drugs. During the fatigue detection task, participants lay on a bed with closed eyes, responding to auditory cues via their headsets by promptly opening their eyes. Volunteers were considered alert if they responded within 2 seconds; otherwise, they were classified as fatigued. Further details regarding the processing of the acquired EEG signals are elaborated in Section 2.1.

This paper utilizes the Hypervolume (HV) metric, as described by While et al. (2006). To evaluate the efficiency of the proposed algorithm, HV measures an algorithm's convergence and diversity by calculating the hypercube's volume, which is formed by the non-dominated solutions obtained by the evaluated algorithm and a predetermined reference point. The reference point is typically chosen to be worse than the function values of each solution in the current evaluated solution set. Therefore, a greater HV value signifies better performance of the assessed algorithm. Since the optimization problem in this paper is a two-objective problem that requires maximization, the reference point  $z$  is generated according to the following Equation (10). Where  $P$  represents the Pareto-optimal-set obtained by the evaluated algorithm.

$$\begin{aligned}
 z &= (z_1, z_2) \\
 z_1 &= \min_{x \in P} f_1(x) - 0.1 \\
 z_2 &= \min_{x \in P} f_2(x) - 0.1
 \end{aligned} \tag{10}$$

As depicted in Figure 6, when conducting channel selection for a BCI system comprising 62 channels, the number of the decision variables is 1,891. Consequently, the channel selection

TABLE 1 Algorithm configuration parameters.

Algorithm	Parameter setting	References
SparseEA2	The crossover probability is 1, while the mutation probability is $1/D$ ;	Zhang et al., 2021
	Both crossover and mutation have a distribution index of 20.	
SLMEA	The crossover probability is 1, while the mutation probability is $1/D$ ;	Tian et al., 2023
	Both crossover and mutation have a distribution index of 20.	
S-ECSO	Inertia weight $w$ is 0.7968; $\mu_{\text{initial}}$ is set to 0.35;	Wang et al., 2022
	The learning factors $C_1$ and $C_2$ are both assigned the value of 1.4962.	
CMMO	The crossover probability is 1, while the mutation probability is $1/D$ ;	Ming et al., 2023
	Both crossover and mutation have a distribution index of 20; $\mu$ , $\tau$ and $\theta$ are set to 0.1.	
S-NSGA-II	The crossover probability is 1, while the mutation probability is $1/D$ ;	Kropp et al., 2023
	Both crossover and mutation have a distribution index of 20.	
TS-MOEA	The crossover probability is 1, while the mutation probability is $1/D$ ;	
	Both crossover and mutation have a distribution index of 20.	
	$R$ , $\mu$ and $s$ are set to 0.1, 0.2 and 40, respectively.	

problem addressed in this paper qualifies as a large-scale MOP. Moreover, owing to the sparsity of the correlation matrix of channels, the channel selection problem discussed in this paper also falls within the category of sparse large-scale MOPs. To assess the effectiveness of the proposed algorithm, TS-MOEA is compared with several advanced large-scale MOEAs, containing SparseEA2 (Zhang et al., 2021), SLMEA (Tian et al., 2023), S-ECSO (Wang et al., 2022), CMMO (Ming et al., 2023), and S-NSGA-II (Kropp et al., 2023). Among these comparison algorithms, SparseEA2 is an effective sparse multi-objective optimization algorithm, whereas S-NSGA-II and S-ECSO are specialized for large-scale multi-objective optimization tasks. CMMO, a newly introduced algorithm, excels in finding an optimal equilibrium between diversity and convergence for multi-objective optimization problems. SLMEA is specialized for super-large-scale sparse multi-objective problems. For fair comparisons, all algorithms adopt the maximum number of function evaluations ( $MaxFE$ ) of 20000 and the population size ( $N$ ) of 200. The detailed settings of algorithms are given in Table 1, and  $D$  is the number of decision variables.

## 4.2 Statistical results and analysis

Table 2 provides the statistical results of all six algorithms over 30 independent runs, measured in terms of HV. In this table, the best average HV values are highlighted in bold. Symbols “+,” “-,” and “ $\approx$ ” indicate that, according to the Wilcoxon rank-sum test (Yaman et al., 2021) at a 5% significance level, the performance of the compared algorithm is significantly better than, worse than, or similar to that of the proposed TS-MOEA, respectively.

Table 2 presents the statistical HV values obtained by TS-MOEA and other MOEAs. The numbers in bold are the best results achieved by algorithms and bold numbers in other tables also indicate the best results. The primary distinction between TS-MOEA and the comparative algorithms lies in TS-MOEA's utilization of a two-stage framework. Within this framework, the

early stage is focused on discovering a diverse and well-distributed population. This is achieved by employing a two-objective problem model that is highly sensitive to the deletion status of channels. The late stage in TS-MOEA directly uses the number of deleted channels as an optimization objective, thereby striking a balance between the number of deleted channels and task accuracy. Furthermore, domain-specific knowledge is utilized to guide the evolutionary process in TS-MOEA. It can be observed from Table 2 that TS-MOEA outperforms other algorithms in terms of HV for all 9 subjects, which indicates the effectiveness of the proposed two-stage framework.

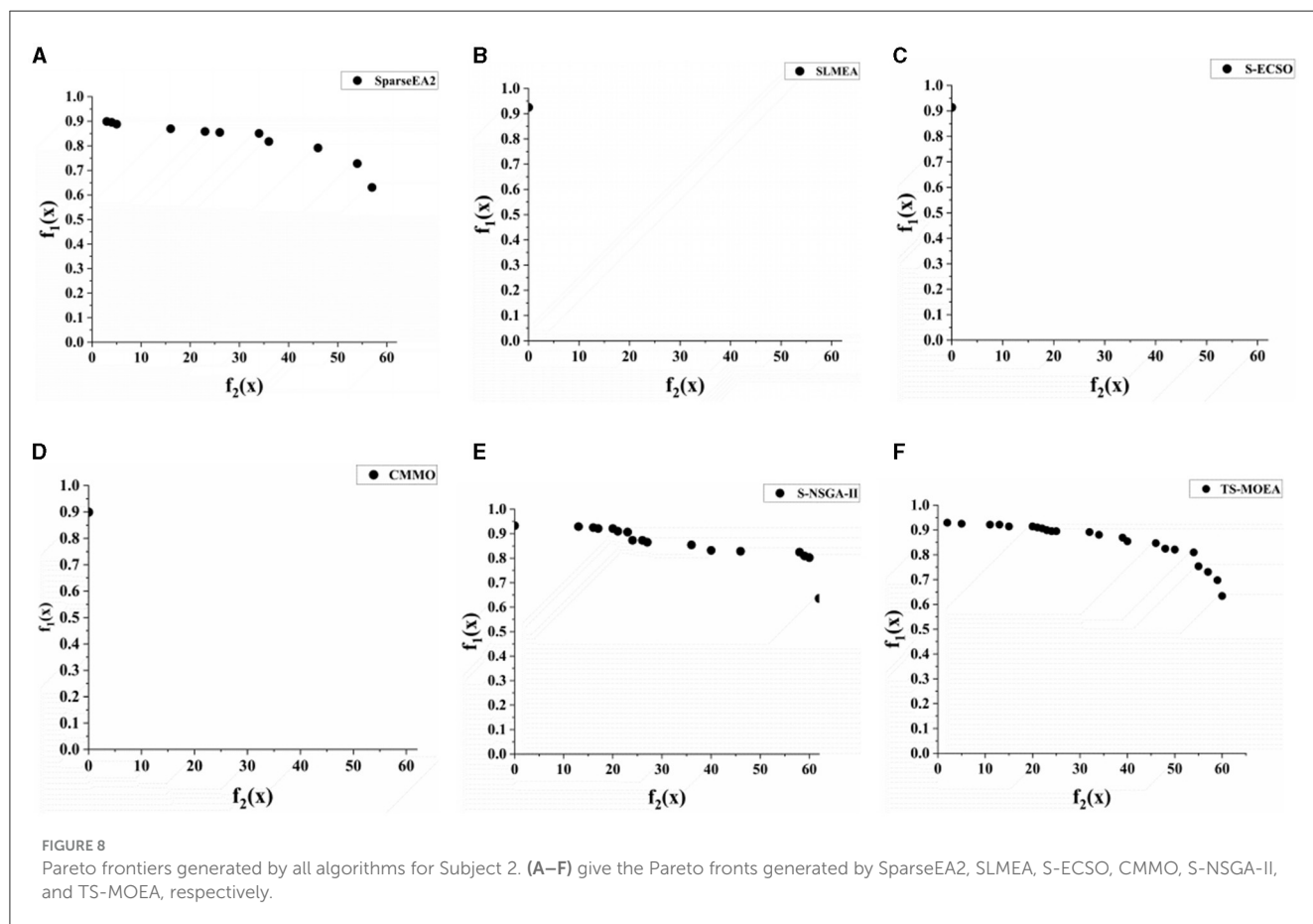
Among the five comparative algorithms, there are algorithms that are specifically designed for sparse large-scale optimization problems. However, these algorithms still perform worse than the proposed algorithm for the channel selection problem. This is mainly because none of these comparative algorithms utilize knowledge related to the problem domain. The statistical results indicate that incorporating domain-specific knowledge into algorithms can effectively enhance their performance when solving specific problems. Figure 8 displays the Pareto fronts generated by all algorithms for Subject 2. In Figure 8, it is evident that compared to other algorithms, TS-MOEA obtains the best Pareto front, which also verifies the efficiency of the proposed algorithm.

Table 3 gives the average classification accuracies achieved by SVM using all channels and partial channels selected by TS-MOEA for all subjects. Since TS-MOEA provides a set of Pareto optimal solutions, which includes a variety of different electrode selection schemes. To save space, Table 3 displays several representative channel selection schemes along with their corresponding classification accuracies. As can be seen from Table 3, the classification accuracy decreases as the number of selected channels decreases. However, in some cases where only a subset of channels is chosen (such as selecting 60 or 52 channels), the classification accuracy is either better than or slightly lower than when all channels are selected. This indicates that the proposed TS-MOEA can effectively reduce the



TABLE 2 Average HV values achieved by TS-MOEA and other comparative MOEAs.

Subject	SparseEA2	SLMEA	S-ECSO	CMMO	S-NSGA-II	TS-MOEA
1	2.79E+01 -	4.46E+00 -	1.36E+01 -	8.86E+00 -	5.13E+00 -	<b>6.16E+02</b>
2	2.42E+01 -	1.25E+01 -	1.81E+01 -	2.15E+01 -	1.16E+02 -	<b>6.63E+02</b>
3	4.51E+01 -	7.01E+00 -	1.99E+01 -	1.26E+01 -	1.17E+02 -	<b>7.41E+02</b>
4	2.11E+01 -	4.63E - 01 -	9.35E+00 -	6.58E+01 -	1.04E+02 -	<b>4.29E+02</b>
5	2.93E+01 -	3.02E+01 -	2.42E+01 -	2.03E+01 -	3.13E+01 -	<b>7.33E+02</b>
6	2.21E+01 -	1.30E+01 -	9.70E+00 -	7.36E+00 -	1.32E+01 -	<b>6.28E+02</b>
7	2.02E+01 -	9.73E+00 -	1.13E+01 -	6.21E+00 -	1.68E+01 -	<b>7.27E+02</b>
8	2.73E - 01 -	9.99E+00 -	5.20E+01 -	1.44E+00 -	1.13E+02 -	<b>3.81E+02</b>
9	2.08E+01 -	1.28E+01 -	1.13E+01 -	5.13E+00 -	1.06E+02 -	<b>5.37E+02</b>
+/-/≈	0/9/0	0/9/0	0/9/0	0/9/0	0/9/0	



number of channels used in BCIs while maintaining acceptable classification accuracy.

For further comparison of the proposed TS-MOEA with other state-of-the-art MOEAs, including SparseEA2, SLMEA, S-ECSO, CMMO, and S-NSGA-II, Table 4 presents the average classification accuracies achieved for all subjects based on the varying number of channels selected by the algorithm. It can be observed from Table 4, SLMEA, S-ECSO, CMMO fail to provide classification accuracies in most cases. This is due to

the poor diversity of these three algorithms (which can also be observed in Figures 8B–D), which results in their inability to obtain the Pareto-optimal solutions for the corresponding number of selected channels. SparseEA2 and S-NSGA-II are capable of obtaining well-distributed Pareto optimal solution sets, similar to the proposed TS-MOEA. However, in terms of classification accuracy, TS-MOEA achieves the best results. Therefore, the statistical results in Table 4 validate that the proposed algorithm indeed strikes a good balance between classification accuracy and

the number of selected channels compared to the algorithms under comparison.

Figure 9 demonstrates the convergence of six comparative algorithms on 9 subjects. It can be observed from Figure 9 that TS-MOEA exhibits faster convergence compared to the other algorithms. This is mainly because the problem model adopted in the early stages of TS-MOEA can effectively prevent stagnation in the search process. Additionally, the sparse initialization and Score-based mutation operators can also accelerate the convergence speed of TS-MOEA.

In summary, there are two reasons for the superior performance of the proposed algorithm. First, TS-MOEA adopts a two-stage framework with different optimization models for each stage, which maintains population diversity and avoids premature maturity of the algorithm. It can be observed from Figure 8 that TS-MOEA has obtained a Pareto front with a better distribution. Second, in TS-MOEA, operators related to the problem domain are used to improve the effectiveness of the searching process. Specifically, when assigning scores to each decision variable, the position of the channels and the distance between them are considered. As shown in Figure 9, TS-MOEA demonstrates the best convergence, which also indicates the effectiveness of the domain-related operators used in TS-MOEA.

Figure 10 presents the average execution time of all algorithms tested on Subject 1. It can be found from Figure 10 that S-NSGA-II has the least running time, followed by E-ECSO and the proposed

TS-MOEA. SparseEA2 has the highest average running time. This is because the strategy used in SparseEA2 to obtain the Score values of decision variables incurs an increased computational cost, especially when there is a significant amount of decision variables. In S-NSGA-II, operators designed for efficient handling of large-scale sparse multi-objective optimization problems are introduced. These operators enable S-NSGA-II to achieve high efficiency when dealing with channel selection problems with a significant amount of decision variables. For S-ECSO, the algorithm employs a three-party competition mechanism to guide its evolutionary process. Compared to commonly used genetic operators such as crossover and mutation, the three-party competition mechanism is simpler and consumes less computational cost.

### 4.3 Statistical results and analysis on the DEAP dataset

The DEAP (Koelstra et al., 2011) dataset was collected from a group of 32 participants, specifically for human emotion recognition. The dataset consisted of 32 channels of EEG signals and 8 channels of peripheral physiological signals (PPS.) The signals were sampled at a rate of 512 Hz. During the data collection process, the participants watched 40 1-min music videos while their physiological signals were recorded. The data set for each trial consisted of a 3-second pre-trial time and a 60-second video viewing trial time. At the end of the trial, participants self-assessed themselves based on arousal, sense of worthiness, dominance, and likability, using discrete 9-point scales for each dimension. In this section, the arousal dimension from the DEAP dataset has been utilized as a categorical label for binary classification to validate the effectiveness of the proposed algorithm.

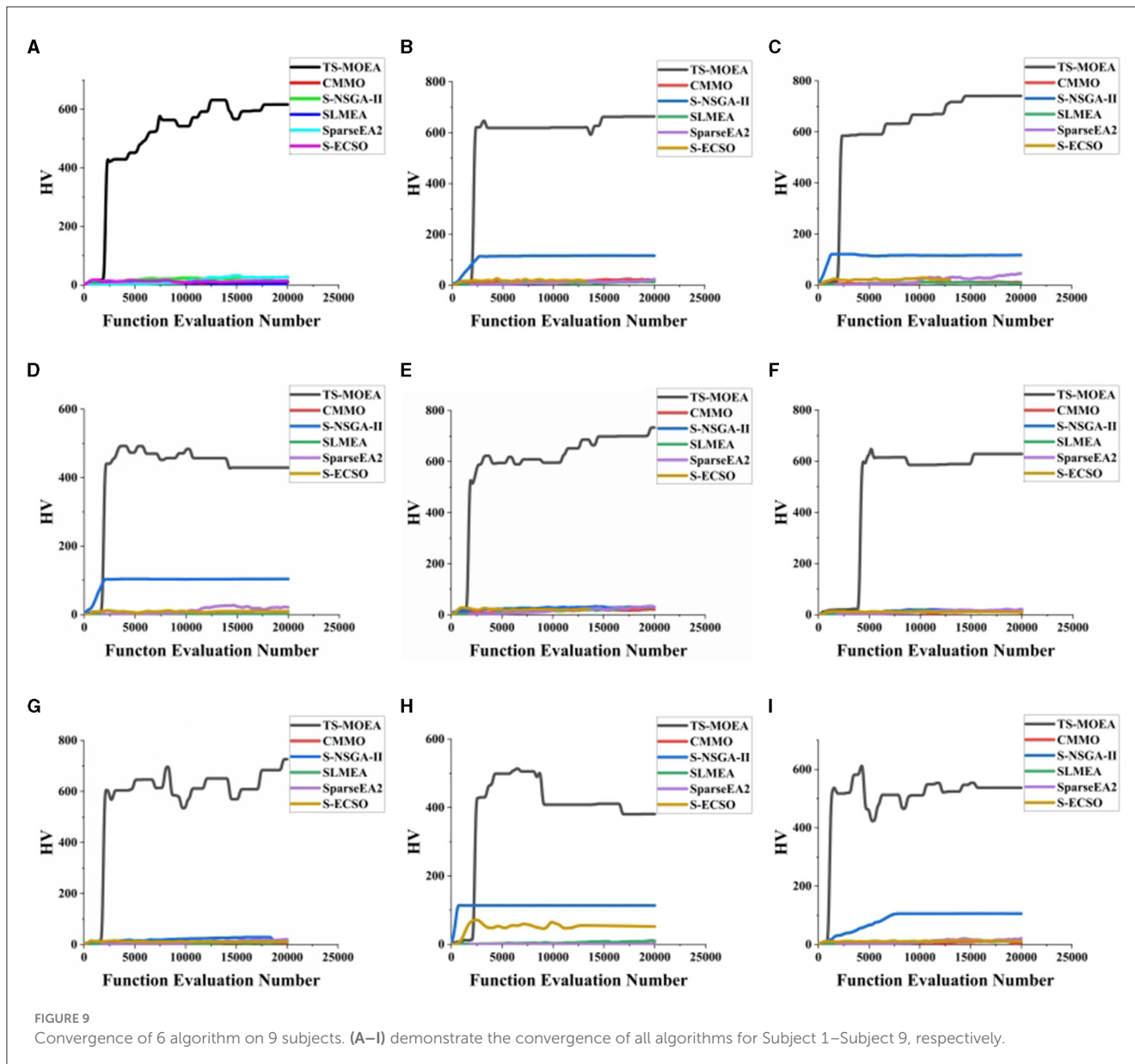
Table 5 demonstrates the average classification accuracy of the proposed algorithm on DEAP. As can be seen from Table 5, when using 30, 22, 17, and 12 channels, the average classification accuracy is close to the classification accuracy using all channels, which further demonstrates the effectiveness of the proposed algorithm. At the same time, the results also show that channel selection is not simply a matter of reducing the number of channels to improve the classification performance, and that it is necessary to find the optimal combination of channels.

TABLE 3 Average classification accuracies achieved by SVM using all channels and partial channels selected by TS-MOEA.

SVM using channels selected by TS-MOEA		SVM using all channels
Number of selected channels	Accuracy rate	
60	98.62%	98.53%
52	95.18%	
42	88.02%	
32	80.55%	
22	70.27%	
12	57.09%	
2	53.23%	

TABLE 4 Average classification accuracies obtained by all algorithms.

Number of selected channels	SparseEA2	SLMEA	S-ECSO	CMMO	S-NSGA-II	TS-MOEA
60	91.50%	97.48%	-	-	60.36%	98.62%
52	76.78%	-	-	-	58.22%	95.18%
42	60.37%	-	-	-	56.47%	88.02%
32	56.92%	-	-	-	56.47%	80.55%
22	56.47%	-	-	-	52.19%	70.27%
12	55.40%	-	-	-	51.43%	57.09%
2	45.98%	-	-	-	51.43%	53.23%



## 5 Discussion

### 5.1 Investigate of the zero assignment parameter $s$

As shown in Line 10 of Algorithm 1,  $s$  is modulates the difficulty level in deleting channels. Specifically, for the  $k^{th}$  sample, if the correlation coefficients between the  $j^{th}$  channel and more than  $s$  other channels are 0 in the filtered correlation matrix, then the  $j^{th}$  channel can be deleted. In this case, Setting  $s$  to a large value makes it challenging to meet the channel deletion criteria, potentially leading the algorithm into stagnation. Conversely, a small  $s$  might result in erroneous deletion of channels.

In this section, Subjects 1, 5, 6, 7, and 9 are chosen to analyze the impact of different  $s$  values on the effectiveness of the proposed TS-MOEA.  $s$  takes values within the range of [35, 55] with a step size of 5. The average HV values across 30 independent runs for

different  $s$  on the selected five subjects can be found in Table 6. It can be observed that  $s = 40$  achieves the highest average HV values. Taking the number of selected channels as 2, 12, 22, 32, 42, 52, and 60 as examples, Table 7 provides the average classification accuracies of TS-MOEA for different values of  $s$ . As shown in Table 7, when  $s = 40$ , the algorithm achieves the optimal classification accuracy for most of the lead selection schemes. Hence, in this paper,  $s$  takes the value of 40.

### 5.2 Investigate of the transition control parameter $\mu$

It can be observed in Algorithm 2 that the parameter  $\mu$  governs the transition from the early stage to the late stage in TS-MOEA. As described in Line 4 of Algorithm 2, the

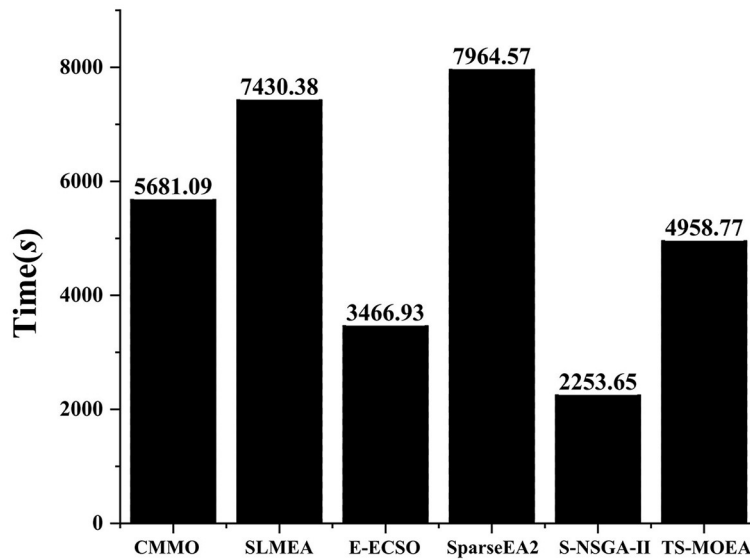


FIGURE 10 Average running times for all algorithms on Subject 1.

TABLE 5 Average classification accuracies on DEAP achieved by SVM using all channels and partial channels selected by TS-MOEA.

SVM using channels selected by TS-MOEA		SVM using all channels
Number of selected channels	Accuracy rate	
30	72.08%	73.18%
27	69.38%	
22	71.88%	
17	71.63%	
12	71.25%	
7	66.09%	
2	60.01%	

algorithm will shift from the early stage to the late stage if the number of function evaluations consumed by the early stage exceeds the predefined  $t, \mu \times MaxFE$ , or if the obtained Pareto-optimal solutions satisfy the diversity requirement. Therefore, if  $\mu$  is set too large, the algorithm may exhaust a significant number of function evaluations in the early stage, potentially leading to unnecessary computational waste. Conversely, if  $\mu$  is too small, the early stage might fail to produce solutions with a good distribution. Consequently, the late stage may not achieve satisfactory optimization results, given that the solutions obtained in the early stage serve as initial solutions in the late stage.

In this section, the influence of different  $\mu$  on TS-MOEA's performance is investigated using five selected subjects: Subjects 1, 5, 6, 7, and 9.  $\mu$  is within the interval  $[0, 1]$  using an increment of

0.2. When  $\mu = 0$ , TS-MOEA exclusively executes the early stage, while  $\mu = 1$  means that only the late stage is executed. Table 8 provides the average HV values obtained from 30 independent runs for different  $\mu$  values across the selected five volunteers. Statistical results indicate that the performance of TS-MOEA that operates in only one stage ( $\mu = 0$  and  $\mu = 1$ ) is inferior to that of the algorithm utilizing both stages concurrently ( $\mu = 1/5, \mu = 2/5, \mu = 3/5$ , and  $\mu = 4/5$ ). As shown in Table 8,  $\mu = 1/5$  achieves the best results for 2 out of 5 subjects. However,  $\mu = 1/5$  also obtains the optimal mean values across all selected subjects. Table 9 gives the average classification accuracies of different  $\mu$  for all subjects and  $\mu = 1/5$  achieves the best performance for most cases. Therefore,  $\mu$  is set to 1/5 in this paper.

### 5.3 Investigation of the distance radius R

In TS-MOEA, the distance radius  $R$  is adopted to calculate the scores of decision variables as depicted in Equation (9). If the distance between two channels is less than  $R$ , the score value assigned to these channels will be small, leading to a corresponding small value in the decision variable. Consequently, the correlation coefficient between the aforementioned channels is less likely to become 0 after filtering (Algorithm 4). If  $R$  is set to a small value, the probability of considering two channels as unrelated becomes low. In this case, the algorithm might retain channels that are useless to the specific task.

In this study, the distance radius  $R$  varies from 0.2 to 2 with increments of 0.2, as the maximum distance between two channels is 2. Figure 11 displays the average HV values obtained by TS-MOEA with different  $R$  across all subjects by 30 independent runs. It is evident from Figure 11 that the best performance is achieved when  $R = 1.0$ . Table 10 gives the average classification accuracies



TABLE 6 Average HV values for different  $s$ .

Subject \ $s$	35	40	45	50	55
Subject 1	6.62E+02	6.62E+02	5.25E+02	4.05E+02	6.39E+02
Subject 5	6.02E+02	6.56E+02	6.02E+02	3.37E+02	3.94E+02
Subject 6	6.28E+02	6.95E+02	6.26E+02	5.05E+02	4.86E+02
Subject 7	5.70E+02	6.39E+02	5.94E+02	4.61E+02	5.28E+02
Subject 9	4.47E+02	4.45E+02	4.82E+02	3.27E+02	3.74E+02
Mean	5.82E+02	6.14E+02	5.66E+02	4.07E+02	3.69E+02

TABLE 7 Average classification accuracies of different  $s$  for all subjects.

Number of selected channels	$s$				
	35	40	45	50	55
60	98.40%	99.20%	98.14%	98.78%	99.18%
52	98.40%	98.50%	97.53%	97.56%	97.32%
42	96.70%	97.30%	95.61%	95.90%	97.87%
32	95.59%	95.73%	95.12%	93.85%	95.67%
22	91.92%	88.63%	84.88%	90.18%	93.07%
12	77.89%	79.13%	77.00%	77.10%	78.46%
2	77.27%	78.13%	77.00%	76.98%	77.23%

TABLE 8 Average HV values for different  $\mu$ .

Subject \ $\mu$	0	1/5	2/5	3/5	4/5	1
Subject 1	7.85E+02	6.21E+02	5.95E+02	6.34E+02	5.87E+02	2.12E+01
Subject 5	1.61E+01	6.57E+02	6.22E+02	6.40E+02	5.82E+02	1.51E+01
Subject 6	1.64E+01	5.30E+02	5.69E+02	5.39E+02	5.24E+02	1.88E+01
Subject 7	3.70E+02	5.49E+02	5.60E+02	5.84E+02	5.40E+02	1.69E+01
Subject 9	1.22E+01	5.61E+02	4.20E+02	4.77E+02	4.38E+02	1.18E+01
Mean	2.40E+02	5.83E+02	5.53E+02	5.75E+02	5.34E+02	1.68E+01

of different  $R$  for all subjects and  $R = 1.0$  achieves the best performance for most cases. Therefore, the distance radius  $R$  is set to 1.0 in this paper.

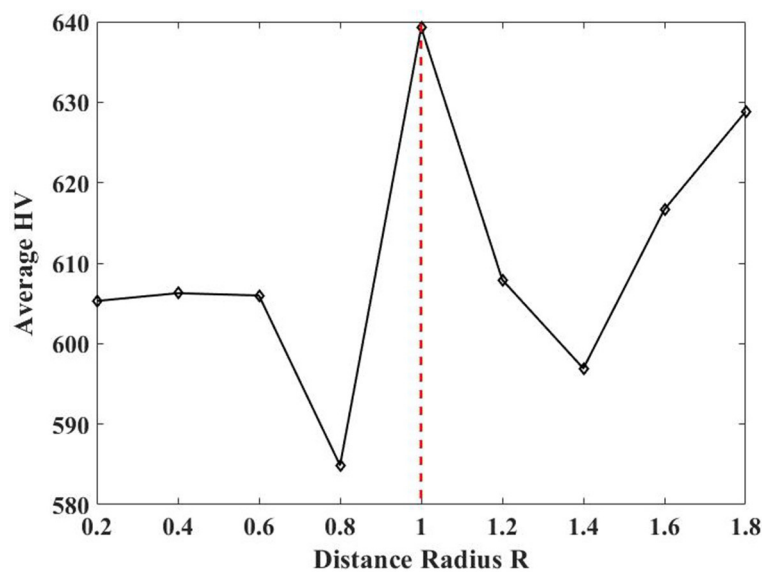
## 5.4 Investigation of the selected channels

This section discusses the channels selected by the proposed TS-MOEA, using the fatigue detection task in Section 4.2 as an example. Figure 12 illustrates the scenarios with selected numbers of channels at 52, 42, 32, 22, 12, and 2 for all subjects, and Table 3 shows the corresponding average classification accuracies for the six cases. As shown in Figure 12 and Table 3, the average classification accuracy gradually decreases as the number of deleted channels increases. It can be observed that the channels selected from Figures 12A–D essentially include the frontal (Fpz, F3, Fz, F6),

frontotemporal (FT7, Fc6, FT8), and central (C5, C6, T8) regions. Specific activity patterns in the frontotemporal region may be associated with dreams and cognitive activity during sleep. Activity in the central regions may be associated with motor inhibition and somatosensory information processing during sleep, and activity patterns in these regions may reflect changes in muscle relaxation and sensory information transfer during sleep. From Figures 12D–F, the aforementioned ten channels were deleted, and there is a noticeable decline in classification accuracy, which can be seen in Table 3. Therefore, the removal of channels from regions closely related to the fatigue detection task will result in a sharp decline in classification accuracy. From this, it can be understood that incorporating prior knowledge of regions of interest (ROIs) related to specific tasks into the lead selection algorithm may be beneficial. For instance, in the context of fatigue detection tasks as discussed in Section 4.2, prioritizing the retention of channels from the frontal, frontotemporal, and central regions can help the channel selection

TABLE 9 Average classification accuracies of different  $\mu$  for all subjects.

Number of selected channels	$\mu$					
	0	1/5	2/5	3/5	4/5	1
60	91.98%	<b>92.75%</b>	91.66%	92.32%	90.98%	90.66%
52	90.82%	90.47%	90.50%	<b>90.86%</b>	89.20%	90.50%
42	<b>89.28%</b>	89.14%	88.25%	88.92%	87.97%	88.25%
32	86.57%	<b>88.57%</b>	87.62%	86.94%	86.05%	87.42%
22	85.46%	<b>86.14%</b>	85.55%	86.57%	85.42%	85.55%
12	83.23%	<b>84.36%</b>	83.47%	83.71%	84.02%	83.47%
2	70.15%	<b>76.85%</b>	75.59%	76.29%	74.47%	75.59%

FIGURE 11  
Average HV values of different R.

algorithm strike a balance between the number of leads chosen and the classification accuracy.

## 6 Conclusions

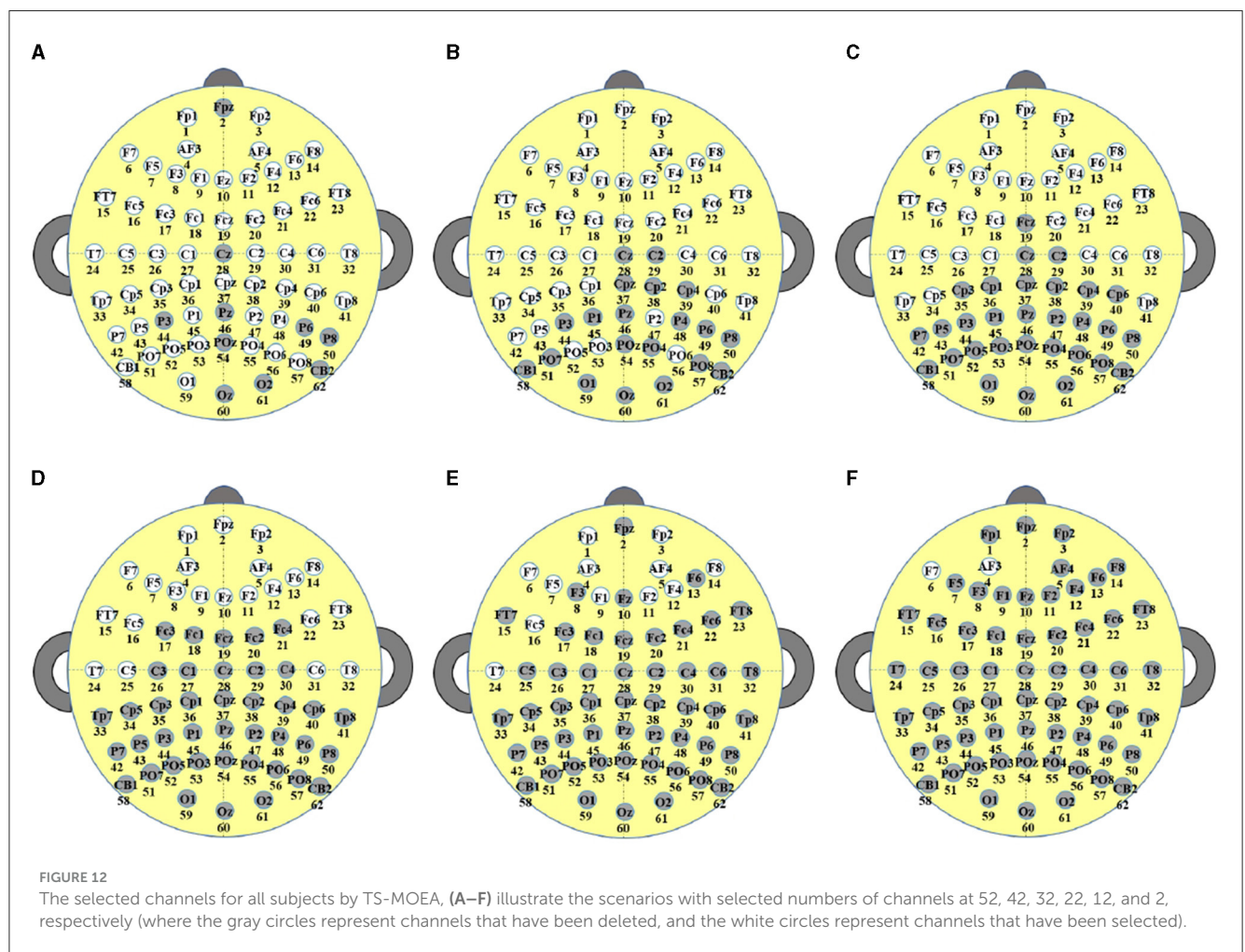
This paper introduces a two-stage sparse multi-objective evolutionary algorithm (TS-MOEA) to solve channel selection problems within BCIs. In TS-MOEA, a two-stage framework with two different two-objective problem models has been adopted. Specifically, a two-objective problem which is sensitive to channel deletion is used in the early stage of TS-MOEA to prevent the algorithm from stalling. In the late stage of TS-MOEA, a two-objective problem model that can directly indicate the number of deleted channels is utilized. To strike a balance between convergence and population diversity in TS-MOEA, a transition condition has been devised. This condition takes into account both the number of consumed function evaluations and the distribution of the current population to control the transition

between the early and late stages of the proposed algorithm. Moreover, due to the sparsity of the correlation matrix of channels, a sparse initialization operator is introduced to generate the initial population. Furthermore, a *Score*-based mutation operator has been integrated to enhance the search efficiency of the early stage in TS-MOEA. The experimental results of TS-MOEA and five other advanced MOEAs have demonstrated the efficiency of the proposed algorithm. However, as shown in Figure 8, Tables 3, 5, TS-MOEA provides a set of Pareto-optimal solutions, each offering a different channel selection scheme. Therefore, TS-MOEA does not directly yield a single optimal electrode selection scheme; in practical applications, the user must make a decision on which channel selection scheme to choose from the Pareto-optimal solution set. Additionally, although TS-MOEA takes into account knowledge relevant to the problem, such as channel positions and the distance matrix between channels, it does not consider the impact of regions of interest (ROI) on the performance of the algorithm.

As shown in Section 3.2, TS-MOEA incorporates the problem-domain knowledge, specifically the locations and distance matrix

TABLE 10 Average classification accuracies of different R for all subjects.

Number of selected channels	R								
	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8
60	91.57%	92.55%	91.79%	92.18%	<b>92.58%</b>	91.35%	92.07%	92.79%	92.05%
52	90.13%	90.85%	90.86%	90.85%	<b>91.82%</b>	90.91%	91.61%	91.40%	91.13%
42	89.57%	90.05%	89.75%	89.94%	<b>90.63%</b>	89.37%	90.36%	90.42%	89.15%
32	87.94%	88.26%	<b>88.67%</b>	87.86%	88.50%	88.19%	87.67%	88.17%	86.70%
22	<b>86.40%</b>	85.40%	84.95%	84.68%	86.04%	84.63%	84.20%	84.63%	84.90%
12	<b>82.52%</b>	80.66%	82.08%	79.98%	82.36%	80.97%	81.17%	81.13%	79.64%
2	71.80%	63.43%	73.88%	65.57%	<b>74.40%</b>	63.45%	63.40%	63.06%	65.93%



of channels, to enhance the algorithm’s performance. However, the biological connections between brain regions, which could better capture and exploit correlations between different channels, were not considered. Therefore, how to combine the biological connections between brain regions in the design of critical operators to improve the search capabilities of the algorithm is one of the future works of this paper. Furthermore, as the number of commands increases, the number of brain wave patterns (or other physiological signals) that the BCI system needs to distinguish becomes larger, which increases the complexity of the classification

task. Therefore, how to maintain or improve classification accuracy when the number of commands increases will be one of the future works of this paper.

### Data availability statement

The data analyzed in this study is subject to the following licenses/restrictions: the data that support the findings of this study are available on request from the corresponding author upon

reasonable request. Requests to access these datasets should be directed to [liuty@shmtu.edu.cn](mailto:liuty@shmtu.edu.cn).

## Ethics statement

The studies involving human participants were reviewed and approved by the Medical and Life Science Ethics Committee of Tongji University. Written informed consent to participate in this study was provided by the participants.

## Author contributions

TL: Conceptualization, Formal analysis, Funding acquisition, Investigation, Methodology, Supervision, Validation, Writing—original draft, Writing—review & editing. YW: Data curation, Methodology, Formal analysis, Software, Visualization, Writing—review & editing. AY: Conceptualization, Data curation, Methodology, Software, Visualization, Writing—original draft, Writing—review & editing. LC: Data curation, Funding acquisition, Validation, Writing—original draft. YC: Data curation, Writing—review & editing.

## References

- Abdullah, Faye, I., and Islam, M. R. (2022). EEG channel selection techniques in motor imagery applications: a review and new perspectives. *Bioengineering* 9:726. doi: 10.3390/bioengineering9120726
- Almanza-Conejo, O., Avina-Cervantes, J. G., Garcia-Perez, A., and Ibarra-Manzano, M. A. (2023). A channel selection method to find the role of the amygdala in emotion recognition avoiding conflict learning in EEG signals. *Eng. Applic. Artif. Intell.* 126:106971. doi: 10.1016/j.engappai.2023.106971
- Alotaiby, T., El-Samie, F. E. A., Alshebeili, S. A., and Ahmad, I. (2015). A review of channel selection algorithms for EEG signal processing. *EURASIP J. Adv. Signal Proc.* 2015, 1–21. doi: 10.1186/s13634-015-0251-9
- Aydore, S., Pantazis, D., and Leahy, R. M. (2013). A note on the phase locking value and its properties. *NeuroImage* 74, 231–244. doi: 10.1016/j.neuroimage.2013.02.008
- Chen, Z., Duan, S., and Peng, Y. (2022). EEG-based emotion recognition by retargeted semi-supervised regression with robust weights. *Systems* 10:236. doi: 10.3390/systems10060236
- Deb, K., and Beyer, H.-G. (2001). Self-adaptive genetic algorithms with simulated binary crossover. *Evolut. Comput.* 9, 197–221. doi: 10.1162/106365601750190406
- Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Trans. Evolut. Comput.* 6, 182–197. doi: 10.1109/4235.996017
- Deist, T. M., Grewal, M., Dankers, F. J. W. M., Alderliesten, T., and Bosman, P. A. N. (2023). “Multi-objective learning using hv maximization,” in *Evolutionary Multi-Criterion Optimization*, eds. M. Emmerich, A. Deutz, H. Wang, A. V. Kononova, B. Naujoks, K. Li, et al. (Cham: Springer Nature Switzerland), 103–117. doi: 10.1007/978-3-031-27250-9\_8
- Deng, W., Zhang, X., Zhou, Y., Liu, Y., Zhou, X., Chen, H., et al. (2022). An enhanced fast non-dominated solution sorting genetic algorithm for multi-objective problems. *Inf. Sci.* 585, 441–453. doi: 10.1016/j.ins.2021.11.052
- do Val Lopes, C. L., Martins, F. V. C., Wanner, E. F., and Deb, K. (2022). Analyzing dominance move (mip-dom) indicator for multiobjective and many-objective optimization. *IEEE Trans. Evolut. Comput.* 26, 476–489. doi: 10.1109/TEVC.2021.3096669
- Ganesh, N., Shankar, R., Kalita, K., Jangir, P., Oliva, D., and Pérez-Cisneros, M. (2023). A novel decomposition-based multi-objective symbiotic organism search optimization algorithm. *Mathematics* 11:1898. doi: 10.3390/math11081898
- Gong, H., Li, Y., Zhang, J., Zhang, B., and Wang, X. (2024). A new filter feature selection algorithm for classification task by ensembling pearson correlation coefficient and mutual information. *Eng. Applic. Artif. Intell.* 131:107865. doi: 10.1016/j.engappai.2024.107865
- He, L., Chiong, R., Li, W., Budhi, G. S., and Zhang, Y. (2022). A multiobjective evolutionary algorithm for achieving energy efficiency in production environments integrated with multiple automated guided vehicles. *Knowl. Based Syst.* 243:108315. doi: 10.1016/j.knsys.2022.108315
- Ishibuchi, H., Imada, R., Masuyama, N., and Nojima, Y. (2019). “Comparison of hypervolume, igd and igd+ from the viewpoint of optimal distributions of solutions,” in *Evolutionary Multi-Criterion Optimization*, eds. K. Deb, E. Goodman, C. A. Coello Coello, K. Klamroth, K. Miettinen, S. Mostaghim, et al. (Cham: Springer International Publishing), 332–345. doi: 10.1007/978-3-030-12598-1\_27
- Koelstra, S., Muhl, C., Soleymani, M., Lee, J.-S., Yazdani, A., Ebrahimi, T., et al. (2011). Deap: a database for emotion analysis; using physiological signals. *IEEE Trans. Affect. Comput.* 3, 18–31. doi: 10.1109/T-AFCC.2011.15
- Krishna Rao, E., Yaswanthkumar Reddy, N., Greeshma, B., and Vardhan Reddy, Y. S. S. (2022). “EEG based smart wheelchair for disabled persons using non-invasive BCI” in *2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, 440–446. doi: 10.1109/CISES54857.2022.9844334
- Kropp, I., Nejadhashemi, A. P., and Deb, K. (2023). Improved evolutionary operators for sparse large-scale multiobjective optimization problems. *IEEE Trans. Evolut. Comput.* 28, 460–473. doi: 10.1109/TEVC.2023.3256183
- Lavinas, Y., Aranha, C., Sakurai, T., and Ladeira, M. (2018). “Experimental analysis of the tournament size on genetic algorithms,” in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 3647–3653. doi: 10.1109/SMC.2018.00617
- Liu, F., Stephen, E. P., Prerau, M. J., and Purdon, P. L. (2019). “Sparse multi-task inverse covariance estimation for connectivity analysis in EEG source space,” in *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, 299–302. doi: 10.1109/NER.2019.8717043
- Liu, H.-L., Gu, F., and Zhang, Q. (2014). Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems. *IEEE Trans. Evolut. Comput.* 18, 450–455. doi: 10.1109/TEVC.2013.2281533
- Liu, R., Wang, R., Bian, R., Liu, J., and Jiao, L. (2021). A decomposition-based evolutionary algorithm with correlative selection mechanism for many-objective optimization. *Evolut. Comput.* 29, 269–304. doi: 10.1162/evco\_a\_00279
- Liu, T., and Ye, A. (2023). Domain knowledge-assisted multi-objective evolutionary algorithm for channel selection in brain-computer interface systems. *Front. Neurosci.* 17:1251968. doi: 10.3389/fnhum.2024.1400077

## Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported by the National Natural Science Foundation of China (grant Nos. 61806122 and 62102242).

## 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.



- Ma, X., Yu, Y., Li, X., Qi, Y., and Zhu, Z. (2020). A survey of weight vector adjustment methods for decomposition-based multiobjective evolutionary algorithms. *IEEE Trans. Evol. Comput.* 24, 634–649. doi: 10.1109/TEVC.2020.2978158
- Ma, X., Zhang, Q., Tian, G., Yang, J., and Zhu, Z. (2018). On tchebycheff decomposition approaches for multiobjective evolutionary optimization. *IEEE Trans. Evol. Comput.* 22, 226–244. doi: 10.1109/TEVC.2017.2704118
- Maria, M. A., Akhand, M. A. H., Hossain, A. B. M. A., Kamal, M. A. S., and Yamada, K. (2023). A comparative study on prominent connectivity features for emotion recognition from EEG. *IEEE Access* 11, 37809–37831. doi: 10.1109/ACCESS.2023.3264845
- Martinez-Cagigal, V., Santamaria-Vizquez, E., and Hornero, R. (2022). Brain “computer interface channel selection optimization using meta-heuristics and evolutionary algorithms. *Appl. Soft Comput.* 115:108176. doi: 10.1016/j.asoc.2021.108176
- Ming, F., Gong, W., Wang, L., and Gao, L. (2023). Balancing convergence and diversity in objective and decision spaces for multimodal multi-objective optimization. *IEEE Trans. Emerg. Topics Comput. Intell.* 7, 474–486. doi: 10.1109/TETCI.2022.3221940
- Moon, S.-E., Chen, C.-J., Hsieh, C.-J., Wang, J.-L., and Lee, J.-S. (2020). Emotional EEG classification using connectivity features and convolutional neural networks. *Neural Netw.* 132, 96–107. doi: 10.1016/j.neunet.2020.08.009
- Pearson, K. (1895). VII. Note on regression and inheritance in the case of two parents. *Proc. R. Soc. London* 58, 240–242. doi: 10.1098/rsp1.1895.0041
- Pradhan, D., Wang, S., Ali, S., Yue, T., and Liaaen, M. (2021). Cbga-es+: a cluster-based genetic algorithm with non-dominated elitist selection for supporting multi-objective test optimization. *IEEE Trans. Softw. Eng.* 47, 86–107. doi: 10.1109/TSE.2018.2882176
- Premkumar, M., Jangir, P., Sowmya, R., Alhelou, H. H., Heidari, A. A., and Chen, H. (2021). Mosma: multi-objective slime mould algorithm based on elitist non-dominated sorting. *IEEE Access* 9, 3229–3248. doi: 10.1109/ACCESS.2020.3047936
- Qu, R., Wang, Z., Wang, S., Wang, L., Wang, A., and Xu, G. (2023). Intercal electrophysiological source imaging based on realistic epilepsy head model in presurgical evaluation: a prospective study. *Chin. J. Electr. Eng.* 9, 61–70. doi: 10.23919/CJEE.2023.000012
- Reznik, S. J., and Allen, J. J. B. (2018). Frontal asymmetry as a mediator and moderator of emotion: an updated review. *Psychophysiology* 55:e12965. doi: 10.1111/psyp.12965
- Rocha-Herrera, C. A., Diaz-Manriquez, A., Barron-Zambrano, J. H., Elizondo-Leal, J. C., Saldivar-Alonso, V. P., Martinez-Angulo, J. R., et al. (2022). EEG feature extraction using evolutionary algorithms for brain-computer interface development. *Comput. Intell. Neurosci.* 2022:7571208. doi: 10.1155/2022/7571208
- Scannella, S., Pariente, J., De Boissezon, X., Castel-Lacanal, E., Chauveau, N., Causse, M., et al. (2016). N270 sensitivity to conflict strength and working memory: a combined ERP and sloretta study. *Behav. Brain Res. SreeTestContent* 297, 231–240. doi: 10.1016/j.bbr.2015.10.014
- Schreiber, T. (2000). Measuring information transfer. *Phys. Rev. Lett.* 85:461. doi: 10.1103/PhysRevLett.85.461
- Shang, K., and Ishibuchi, H. (2020). A new hypervolume-based evolutionary algorithm for many-objective optimization. *IEEE Trans. Evol. Comput.* 24, 839–852. doi: 10.1109/TEVC.2020.2964705
- Shang, K., Ishibuchi, H., and Ni, X. (2020). R2-based hypervolume contribution approximation. *IEEE Trans. Evol. Comput.* 24, 185–192. doi: 10.1109/TEVC.2019.2909271
- Sibilano, E., Suglia, V., Brunetti, A., Buongiorno, D., Caporusso, N., Guger, C., et al. (2024). *Brain-Computer Interfaces*. New York, NY: Springer US, 203–240. doi: 10.1007/978-1-0716-3545-2\_10
- Tian, Y., Feng, Y., Zhang, X., and Sun, C. (2023). A fast clustering based evolutionary algorithm for super-large-scale sparse multi-objective optimization. *IEEE/CAA J. Autom. Sinica* 10, 1048–1063. doi: 10.1109/JAS.2022.105437
- Tian, Y., Lu, C., Zhang, X., Cheng, F., and Jin, Y. (2022). A pattern mining-based evolutionary algorithm for large-scale sparse multiobjective optimization problems. *IEEE Trans. Cyber.* 52, 6784–6797. doi: 10.1109/TCYB.2020.3041325
- Tian, Y., Lu, C., Zhang, X., Tan, K. C., and Jin, Y. (2021a). Solving large-scale multiobjective optimization problems with sparse optimal solutions via unsupervised neural networks. *IEEE Trans. Cybern.* 51, 3115–3128. doi: 10.1109/TCYB.2020.2979930
- Tian, Y., Si, L., Zhang, X., Cheng, R., He, C., Tan, K. C., et al. (2021b). Evolutionary large-scale multi-objective optimization: a survey. *ACM Comput. Surv.* 54, 1–34. doi: 10.1145/3470971
- Tian, Y., Zhang, X., Wang, C., and Jin, Y. (2020). An evolutionary algorithm for large-scale sparse multiobjective optimization problems. *IEEE Trans. Evol. Comput.* 24, 380–393. doi: 10.1109/TEVC.2019.2918140
- van den Broek, S., Reinders, F., Donderwinkel, M., and Peters, M. (1998). Volume conduction effects in EEG and MEG. *Electroencephalogr. Clin. Neurophysiol.* 106, 522–534. doi: 10.1016/S0013-4694(97)00147-8
- Wang, X., Zhang, K., Wang, J., and Jin, Y. (2022). An enhanced competitive swarm optimizer with strongly convex sparse operator for large-scale multiobjective optimization. *IEEE Trans. Evol. Comput.* 26, 859–871. doi: 10.1109/TEVC.2021.3111209
- While, L., Hingston, P., Barone, L., and Huband, S. (2006). A faster algorithm for calculating hypervolume. *IEEE Trans. Evol. Comput.* 10, 29–38. doi: 10.1109/TEVC.2005.851275
- Yaman, A., Iacca, G., Mocanu, D. C., Coler, M., Fletcher, G., and Pechenizkiy, M. (2021). Evolving plasticity for autonomous learning under changing environmental conditions. *Evol. Comput.* 29, 391–414. doi: 10.1162/evco\_a\_00286
- You, Q., Sun, J., Pan, F., Palade, V., and Ahmad, B. (2021). DMO-QPSO: a multi-objective quantum-behaved particle swarm optimization algorithm based on decomposition with diversity control. *Mathematics* 9:1959. doi: 10.3390/math9161959
- Zhang, Q., and Li, H. (2007). MOEA/D: a multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* 11, 712–731. doi: 10.1109/TEVC.2007.892759
- Zhang, Y., Tian, Y., and Zhang, X. (2021). Improved sparseea for sparse large-scale multi-objective optimization problems. *Complex Intell. Syst.* 9, 1127–1142. doi: 10.1007/s40747-021-00553-0
- Zhassuzak, M., Akhmet, M., Amirgaliyev, Y., and Buribayev, Z. (2024). Application of genetic algorithms for periodicity recognition and finite sequences sorting. *Algorithms* 17:101. doi: 10.3390/a17030101
- Zhou, A., Jin, Y., Zhang, Q., Sendhoff, B., and Tsang, E. (2006). “Combining model-based and genetics-based offspring generation for multi-objective optimization using a convergence criterion,” in 2006 *IEEE International Conference on Evolutionary Computation*, 892–899.
- Zille, H., Ishibuchi, H., Mostaghim, S., and Nojima, Y. (2016). “Mutation operators based on variable grouping for multi-objective large-scale optimization,” in 2016 *IEEE Symposium Series on Computational Intelligence (SSCI)* (IEEE), 1–8. doi: 10.1109/SSCI.2016.7850214