Protein Structure Prediction Using A New Optimization-based Evolutionary and Explainable Artificial Intelligence Approach

Jun Hong, *Student Member, IEEE*, Zhi-Hui Zhan, *Fellow, IEEE*, Langchong He, Zongben Xu, Jun Zhang, *Fellow, IEEE*

***Abstract*—Protein structure prediction (PSP) is an important scientific problem because it helps humans to understand how proteins perform their biological functions. This paper models the PSP problem as a multi-objective optimization problem with three fast and accurate knowledge-based energy functions. This way, using evolutionary computation (EC)-based artificial intelligence (AI) approach to solve this multi-objective PSP problem to find the optimal structure is explainable. Considering that the multiple populations for multiple objectives (MPMO) framework shows efficient performance in solving lots of multi-objective benchmarks and real-world problems, this paper proposes a new AI approach named improved MPMO-based differential evolution (IMPMO-DE) to solve the multi-objective PSP problem. To our best knowledge, this is the first time that MPMO is applied to PSP, with three novel strategies. First, an adaptive archive-based mutation strategy is proposed to better balance the exploration and exploitation abilities by adaptively using different archive-based mutation operators in different evolutionary stages. Second, a mixed individual transfer strategy is proposed to share search information among the multiple populations to accelerate the convergence speed. Third, an evolvable archive update strategy is proposed to generate more promising solutions through evolving the archived solutions. IMPMO-DE is tested on 28 representative proteins and all the available template-free modeling proteins up to 404 residues in the famous Critical Assessment of Protein Structure Prediction (CASP14) competition. Experimental results show that IMPMO-DE performs better than the compared state-of-the-art EC-based PSP methods and ranks above average compared with all the CASP14 competitors. More importantly, IMPMO-DE is a new efficient AI approach that opens a promising optimization-based evolutionary and explainable way for efficient PSP rather than deep learning approaches like AlphaFold2, especially for newly discovered proteins without similar known protein structures.**

***Index Terms*—Multi-objective evolutionary algorithm (MOEA), evolutionary computation, artificial intelligence, protein structure prediction (PSP), multiple populations for multiple objectives (MPMO), differential evolution.**

Manuscript received XXXX; revised XXXX; accepted XXXX. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62176094 and Grant U23B2039; and in part by the Tianjin Top Scientist Studio Project under Grant 24JRRCRC00030. (*Corresponding authors: Zhi-Hui Zhan; Jun Zhang.*)

Jun Hong is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, P. R. China.

Zhi-Hui Zhan is with the College of Artificial Intelligence, Nankai University, Tianjin 300350, P. R. China and also with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, P. R. China (e-mail: zhanapollo@163.com).

Langchong He is with the School of Pharmacy, Health Science Center, Xi'an Jiaotong University, Xi'an 710061, P. R. China.

Zongben Xu is with the National Engineering Laboratory for Big Data Analytics, Xi'an Jiaotong University, Xi'an 710049, P. R. China.

Jun Zhang is with the College of Artificial Intelligence, Nankai University, Tianjin 300350, P. R. China and also with the Hanyang University, ERICA, 15588, South Korea.

I. Introduction

PROTEIN is a biological macromolecule that participates in many vital functions, such as delivery of substances, metabolism, and hormone regulation [1]. Protein functions rely on protein structures. Therefore, the determination or prediction of protein structures is usually necessary before exploring protein functions. There are three typical techniques for protein structure determination, which are X-ray diffraction [2], three-dimensional reconstruction (cryo-electron microscopy) [3], and multidimensional nuclear magnetic resonance [4]. The X-ray diffraction technique derives protein structure from the unique diffraction pattern of the protein crystal. The protein purification and crystallization steps in the derivation process are very difficult and time-consuming. The other two techniques, i.e., the three-dimensional reconstruction and the multidimensional nuclear magnetic resonance, require specialized and expensive equipment. The amount of resolved protein structure is far less than the number of known amino acid sequences due to the defects of these determination techniques to some extent. Therefore, how to predict protein structure efficiently and accurately is a scientific problem worth studying and of great practical value.

Researchers have proposed a variety of methods to solve the protein structure prediction (PSP) problem [5]. These methods are mainly divided into two types, one is template-based modeling (TBM) and the other is template-free modeling (TFM). When predicting a protein of unknown structure, TBM method can achieve high prediction accuracy if some proteins with high homology similarity to this protein can be obtained. However, TBM method will not work if the homologous proteins are not available in the protein database. Differently, TFM method does not rely on homologous proteins, but directly predicts protein structure based on the amino acid sequence. Some well-known TFM methods, such as QUARK [6] and Rosetta [7], have achieved efficient performance with the help of Monte Carlo simulations. In recent years, deep learning techniques have also been widely used as TFM methods in solving PSP problems and achieve high prediction accuracies, such as AlphaFold2 [8] and RosettaFold [9].

Although these deep learning-based methods do not need the homologous proteins, they need to use a very large number of known protein structures to train the prediction model.

According to the thermodynamic hypothesis [10], the native structure of protein is at the lowest energy state. Therefore, PSP problem can also be treated as an optimization problem that aims to optimize protein energy functions and find the native structure of protein with the lowest energy. Evolutionary computation (EC) techniques such as genetic algorithm (GA), ant colony optimization (ACO), particle swarm optimization (PSO), and differential evolution (DE), have been successfully applied to solving many complex and large-scale optimization problems [11]-[13]. Many researchers have also used EC techniques to solve the PSP problem through optimizing one protein energy function. These EC-based PSP methods do not rely on protein templates and are TFM methods that can be used to predict newly discovered proteins. Some of these single-objective EC-based PSP methods have shown considerable results, including HGA [14], GLCEA [15], APL [16], HEA [17], ACUE [18], MDE [19], LUE [20], and DPDE [21].

However, due to that proteins have not been completely and clearly understood by human, the energy functions of proteins are often inaccurate, resulting in that the native protein may not be corresponding to the found one with lowest energy state. Therefore, it is necessary to use multiple energy functions to comprehensively measure protein structure from different aspects so as to reduce the deviation. This leads to the multi-objective PSP and has also attracted great research attentions. For example, the improved Pareto archived evolutionary strategy (I-PAES) [22] decomposed chemistry at Harvard macromolecular mechanics (CHARMM) [23] into two objective functions, i.e., bond and non-bond, and solved the PSP problem by optimizing these two objective functions. Their results showed that the prediction accuracy was generally higher than the previously single-objective optimization algorithms. Gao *et al.* [24] proposed a multi-objective algorithm named MO3 that pointed out the importance of solvent-accessible surface area (SASA) to be an energy function. Therefore, MO3 used bond, non-bond, and SASA as three objective functions and achieved prediction accuracy higher than most previously single-objective and two-objective optimization algorithms. Lei *et al.* [25] recently proposed a many-objective algorithm named MO4, which utilized four energy functions as objective functions. Experimental results show that MO4 outperformed all the compared multi-objective algorithms. There have been various kinds of multi-objective PSP methods with promising performance, including MOEA [26], MOEA-PC [27], AIMOES [28], MOPSO [29], and MODE [30].

Traditional multi-objective evolutionary algorithms mainly contain elitist non-dominated sorting GA (NSGA-II) [31] and multi-objective evolutionary algorithm based on decomposition (MOEA/D) [32]. In recent years, researchers have also proposed novel multi-objective EC algorithms, such as EC algorithms based on multiple populations for multiple objectives (MPMO) framework [33]. In MPMO, each population optimizes its corresponding objective, so that the different regions of the Pareto front (PF) can be sufficiently explored. Moreover, the MPMO has an information share mechanism that all the populations share information with each other (e.g. an archive is used to store promising solutions found by different populations and is also shared among all the populations), so that the whole PF can be approached. Zhan *et al.* [33] the first time proposed the novel and efficient MPMO framework and designed a novel coevolutionary multiswarm PSO (CMPSO) algorithm, which was an integration of the MPMO framework and the PSO algorithm. The CMPSO outperforms the traditional NSGA-II algorithm and the MOEA/D algorithm on multiple benchmarks. Later, Wang *et al.* [34] followed the MPMO framework and proposed the cooperative multi-objective DE. Zhang *et al.* [35] applied the MPMO framework to artificial bee colony algorithm. Naidu and Ojha [36] hybridized the MPMO framework with invasive weed optimization. Liang *et al.* [37] proposed a new MPMO framework for constrained multi-objective optimization problems, which designed specific evolutionary strategies by learning the problem types, and this algorithm demonstrated outstanding performance on various problems. Antonio and Coello [38] also made a survey on coevolutionary multi-objective EC and pointed out that MPMO has been a new and novel multi-objective optimization framework. The MPMO-based methods have high effectiveness and efficiency on various benchmarks, and have also been applied to a variety of real-world application problems, including cloud workflow scheduling [39], [40], supply chain optimization [41], airline crew rostering [42], job shop scheduling [43], and logistic scheduling [44], [45]. All these researches show that the MPMO framework has excellent and robust performance in multi-objective and many-objective optimization problem [46], [47].

Based on the above considerations, we establish a multi-objective PSP model and propose an improved MPMO-based DE (IMPMO-DE) algorithm for solving the multi-objective PSP problem. IMPMO-DE extends the MPMO framework through three novel designs, including an adaptive archive-based mutation (AAM) strategy, a mixed individual transfer (MIT) strategy, and an evolvable archive update (EAU) strategy. More specifically, in the AAM strategy, three kinds of archive-based mutation operators, with different exploration and exploitation abilities, are designed to generate new individuals with the guidance of both the populations and the archive. The AAM strategy can adaptively choose one suitable archive-based mutation operator for the current multiple populations, which can better balance the exploration and the exploitation in different evolutionary stages. The MIT strategy is to mix all the individuals and transfer them among all the populations, so as to reallocate individuals in multiple populations that optimize multiple objectives, which can share search information among the multiple populations to accelerate the convergence speed. The EAU strategy can generate more promising solutions through evolving the archived individuals based on random-orient mutation operator and crossover operator. Moreover, the EAU strategy preserves the promising solutions based on the non-dominated sort and crowding distance.

To sum up, the major contributions of this paper include:

1) In the problem modeling aspect, the multi-objective PSP model is established, where three fast and accurate knowledge-based energy functions are elaborately-selected

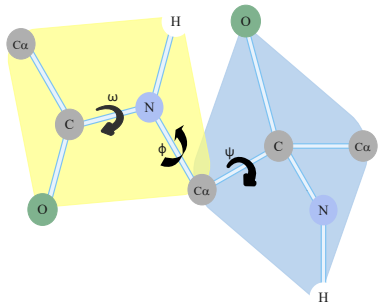


Fig. 1. Torsion angle model of proteins.

and utilized as three objectives. This is helpful to comprehensively evaluate the protein structure from different aspects to evolve to the near-native structure of the protein.

2) In the algorithm designing aspect, the efficient IMPMO-DE algorithm that integrates DE into the improve MPMO framework is proposed to solve the proposed multi-objective PSP model.

3) In the application innovation aspect, the first time, the MPMO framework is applied to solve the PSP problem. It verifies the effectiveness and efficiency of MPMO framework on the PSP problem. More significantly, a new approach has been demonstrated to promote the process of PSP application.

The remainder of this paper includes four parts. Section II introduces the preliminaries of PSP problem and multi-objective optimization. Section III illustrates the proposed IMPMO-DE. Section IV presents the prediction results and analysis between IMPMO-DE and other methods. Section V makes a summarization of this paper.

II. Preliminaries

A. Protein Representation

Proteins are made of amino acids, which contain a great deal of atoms. If each atom is represented in three-dimensional coordinates, then representing a protein will be a very high-dimensional problem, which will be hard for optimization-based approaches to solve it. Moreover, a slight rotation of single bond in protein molecules may lead to huge conformational changes. The rotation of any single bond would lead to a completely different protein conformation. All possible conformations of a protein make up the conformational space of this protein, which is extremely large. How to narrow the conformational search space through protein representation is a significant subproblem in PSP.

Based on the aforementioned considerations, the torsion angle model is established to narrow the conformational search space, as shown in Fig. 1. The single bond -*C*-*N*- has a partial double bond property and it cannot rotate freely. Three adjacent atoms in the peptide chain (i.e., atom *C*, atom *N*, and atom C_α) are in the same plane and this plane is called peptide plane. The bond length and angle between atoms are assumed to be ideal value. Therefore, peptide chain can be regarded as the long chain that consists of a series of peptide plane connecting through the atom C_α. Only the single bond -C-C_α- and the single bond -C_α-N- on the protein backbone can rotate freely. The angle that peptide plane C-N-C_α rotates around the single bond -C_α-N- is called $\boldsymbol{\Phi}$; the angle that peptide plane N-C-C_α rotates around the single bond -C-C_α- is called $\boldsymbol{\psi}$; the angle around the single bond -*C*-*N*- that cannot rotate freely is called $\boldsymbol{\omega}$. These three angles are called torsion angles. Torsion angles $\boldsymbol{\Phi}$ and $\boldsymbol{\psi}$ range in [-180°, 180°], and torsion angle $\boldsymbol{\omega}$ is set as the fixed value 180°. Similarly, sidechains can also be represented with torsion angles and the number of side-chain torsion angles is determined by the type of residues.

Besides, the ranges of torsion angles are limited because of the existence of steric hindrance. The information of protein secondary structures is often utilized to limit the ranges of backbone torsion angles and narrow the feasible conformational space. Furthermore, the ranges of side-chain torsion angles can also be reduced according to the library of rotational isomers dependent on the backbone [48].

B. Fragment Assembly

Fragment assembly is a crucial technique for PSP, which can significantly narrow the search space of protein structures and decrease the computational time [20]. Fragment assembly involves three steps: 1) the amino acid sequence is divided into several fragments. Rosetta, which shows efficient performance in PSP, also uses the fragment assembly technique and sets the fragment length as 3 or 9 [49]. Therefore, this setting is also adopted in this paper; 2) specific fragment library is constructed for each protein. In this paper, the fragment library is constructed through the publicly available ROBETTA full-chain PSP server (http://robetta.bakerlab.org). For a specific protein, the fragment library is generated from the nonhomologous proteins with sequence similarity less than 25%, through the sequence alignment method. For each position in the target sequence, the top 200 fragments with a resolution better than 2 Å are reserved; 3) the fragments of the protein to be predicted are replaced by a randomly selected fragment from the fragment library at the corresponding positions. Therefore, new structures can be obtained through fragment assembly technique and are used in the further optimization. Compared with randomly generated structures, these assembled structures are more feasible mainly due to that tertiary structures of proteins are regular and that nonhomologous proteins may have similar folding patterns.

C. Energy Functions

In order to solve the PSP problem with EC, single or multiple energy functions are selected as objectives to evaluate the energy of protein structure. Energy functions of protein mainly can be classified as physical-based energy functions and knowledge-based energy functions [50]. Physics-based functions rely on molecular dynamics and physical laws, with clear definitions. Some famous physics-based functions, such as CHARMM and AMBER [51], developed in early years and have been widely studied in many early researches [52], [53]. Knowledge-based functions rely on statistical laws of known protein structures. Compared with physics-based functions, knowledge-based functions generally can be calculated faster and the computational resources can be saved. Therefore, various kinds of knowledge-based functions are developed in recent years, such as Rosetta [7], SASA [54], and Rwplus [55]. When EC is used to solve the PSP problem, multiple energy functions are utilized as multiple objectives to comprehensively evaluate

protein structures, mainly for two reasons. One reason is that different kinds of energy functions can measure protein structures from different aspects; another reason is that energy functions may be inaccurate and the native structure of protein may not at the lowest energy state of any single energy function. Abnormal structures may exist if only one energy function is used [56].

Based on the aforementioned considerations, three fast and efficient knowledge-based energy functions are elaborately-selected and utilized as three objectives in this paper. The established multi-objective energy model aims to guide the algorithm to search efficiently in the conformational space that narrowed and constrained by the torsion angle model. Moreover, the physical meaning and mathematical formulas of these three energy functions are described as follows:

1) SASA: The energy function SASA aims to calculate the solvent free energy, to measure the protein surface that the solvent can access. It is assumed that the solvent free energy of each atom can be evaluated by its solvent-accessible surface area. Solvent free energy is significant to correct and help to measure real protein potential energy. Methods that incorporate SASA as one of the objectives have showed efficient performance [24], [25]. Eisenberg *et al.* [54], [57] summarized the formula as

$$G_{sol} = \sum_{i=1} \delta_i \cdot A_i, \tag{1}$$

where the δ_i is the solvation parameters of atom i, A_i is the solvent-accessible surface area of atom i. In this paper, the atomic radius is set to 1.5 angstrom and the energy function SASA is calculated through Pyrosetta [58].

2) Rwplus: Rwplus is an atomic potential that can evaluate and select more promising structures from the decoy. Based on the knowledge of high-resolution protein structures and an ideal random-walk as the reference state, Rwplus has been well-tuned and verified its effectiveness to recognize protein structures that highly related to the native structures [55].

Rwplus consists of two energy terms [59], i.e., the distance-dependent energy term and the orientation-dependent energy term. Rwplus is formulated as

$$\begin{aligned} E_R &= -kT\left[\sum_{\alpha,\beta} \ln \frac{p(\alpha,\beta,R)}{\bar{p}(\alpha,\beta,R)} + 0.1\sum_{A,B} \delta(A,B)\cdot \ln \frac{p(A,B,O_{AB})}{\bar{p}_D(A,B,O_{AB})}\right] \\ &= -kT\left[\sum_{\alpha,\beta} \ln \frac{N(\alpha,\beta,R)}{\bar{N}(\alpha,\beta,R)} + 0.1\sum_{A,B} \delta(A,B)\cdot \ln \frac{N(A,B,O_{AB})}{\bar{N}(A,B,O_{AB})}\right], \end{aligned} \tag{2}$$

where k is the Boltzmann constant, and T is the Kelvin temperature. R represents the Euclidean distance between atom α and atom β. $p(\alpha, \beta, R)$ and $\bar{p}(\alpha, \beta, R)$ represent the observation probability of atom pairs and the expected observation probability respectively. $N(\alpha, \beta, R)$ and $\bar{N}(\alpha, \beta, R)$ are observation quantity of atom pairs and expected observation quantity respectively. (A, B) is a Boolean value and is 1 if vector pairs A and B are in contact, O_{AB} is the relative orientation of vector pairs A and B. Similarly, $p(A, B, R)$ and $\bar{p}(A, B, R)$ represent the observation probability of vector pairs and the expected observation probability respectively. $N(A, B, R)$ and $\bar{N}(A, B, R)$ are observation quantity of vector pairs and expected observation quantity respectively.

3) AACE18: The atom-atom contact energies (AACE) are obtained through maximizing the likelihood of observing the native structures in the non-redundant protein set and are able to recognize the native structures from the decoy structures set [60]. Based on the physicochemical properties of atoms [61], the atoms can be classified into 18 types. The energy function based on these 18-type atoms is named as AACE18.

The energy of AACE18, termed as E, can also be represented as the difference between two terms [62]. One term is the number of atom-water contacts in the extended state, while another one is the number of atom-water contacts in the native state. Therefore, E is defined as

$$\begin{aligned} E &= \sum_{i=1}^{18} e_i \times n_{ir} \\ &= \sum_{i=1}^{18} e_i \times \frac{q_{e,i} \times n_i}{2} - \sum_{i=1}^{18} e_i \times n_{i0} \\ &= e_v \times \left(\sum_{i=1}^{18} \frac{q_{e,i} \times n_i}{2}\right) - e_s \times n_{r0}, \end{aligned} \tag{3}$$

where e_v and e_s are the average energies of the whole atom-water contacts in the extended state and the folded state respectively, and e_i is the average contact energy of atoms belonging to type i. n_i represents the number of atoms that belongs to type i, and n_{ir} represents the total number of solute contacts of atoms that belongs to type i. n_{i0} and n_{r0} represent the number of atom-solvent contacts and solute-solvent contacts respectively. $q_{e,i}$ represents the average number of atoms that are excluded contacted with atoms belonging to type i.

D. Multi-objective Optimization

Based on EC, researchers have used single-objective algorithms or multi-/many-objective algorithms to solve the PSP problem. In general, multi-/many-objective optimization algorithms show higher prediction accuracy than single-objective optimization algorithms. The IMPMO-DE proposed in this paper is also a multi-objective optimization algorithm that considers three objectives, i.e., SASA, RWplus, and AACE18.

Multi-objective optimization algorithms optimize two or more objectives simultaneously [63]. However, the optimization of some objectives often leads to the deterioration of other objectives due to the conflicting relationship among different objectives. Mathematically, a multi-objective minimization problem can be expressed as

$$\begin{aligned} &Minimize\ f(\boldsymbol{x}) = \left\{f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), ... f_M(\boldsymbol{x})\right\} \\ &Subject\ to\ \boldsymbol{x} = \left(x_1, x_2, ..., x_n\right) \in R^n, \end{aligned} \tag{4}$$

where $\boldsymbol{x}$ is a decision vector in the decision space R^n and $f(\boldsymbol{x})$ is the objective vector in the objective space R^M. n represents the dimension of decision variables, and M represents the number of objective functions.

Moreover, Pareto dominance is used to compare the quality of two decision vectors due to that the objectives are conflict. For a minimization problem, suppose that $\boldsymbol{a}$ and $\boldsymbol{b}$ are two feasible solutions in R^n, $\boldsymbol{a}$ dominates $\boldsymbol{b}$ if and only if

$$\begin{aligned} &\forall i \in \left\{1, 2, ..., M\right\}, f_i(\boldsymbol{a}) \le f_i(\boldsymbol{b}) \\ &\exists j \in \left\{1, 2, ..., M\right\}, f_j(\boldsymbol{a}) < f_j(\boldsymbol{b}). \end{aligned} \tag{5}$$

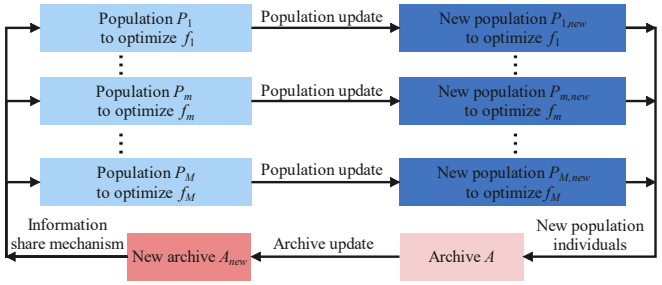


Fig. 2. The major structure of MPMO framework.

The feasible solution $\boldsymbol{x}^*$ is Pareto optimal solution, if and only if any decision vector in R^n cannot dominate $\boldsymbol{x}^*$. All the Pareto optimal solutions in R^n form the Pareto optimal solution set. The Pareto optimal solution set is formulated as

$$P = \left\{ \boldsymbol{x}^* \in R^n \mid \neg \exists \boldsymbol{x} \in R^n, \boldsymbol{x} \text{ dominates } \boldsymbol{x}^* \right\}. \quad (6)$$

The Pareto optimal solution set can be mapped to the objective space through the objectives and form the PF, i.e.

$$PF = \left\{ f(\boldsymbol{x}) \mid \boldsymbol{x} \in P \right\}. \quad (7)$$

A variety of multi-objective evolutionary algorithms have been proposed to deal with the multi-objective optimization problems. However, most of these algorithms treat multiple objectives as a whole and it is hard for these algorithms to evaluate and select individuals because that different objectives generally conflict with each other. MPMO-based methods deal with this difficulty by associating each population with only one objective, and the individuals in each population are evaluated and selected based on this corresponding objective. This is the major superiority of MPMO-based methods compared with conventional multi-objective algorithms.

The major structure of the MPMO framework is shown as Fig. 2. The individuals in the same population are compared based on only one objective that is associated with this population. So that, the individuals will not be confused by different objectives with conflicts, but are able to search different regions of the PF under the guidance of the corresponding objective. However, since each population only focuses on optimizing its single objective, the individuals in the same population may converge to the extreme point of the corresponding objective, which may result in that the PF not being fully explored. In order to solve this issue, the MPMO framework also has the information share mechanism that an archive is used to store the Pareto optimal individuals found by all the populations and is also shared among all the populations. New population individuals generated by multiple populations are also inserted into the archive, and the archive is updated to maintain the non-dominated solutions.

III. IMPMO-DE FOR PSP

A. Overview of IMPMO-DE

In this paper, PSP is modeled as a multi-objective problem and the IMPMO-DE algorithm is proposed to solve it. To our best knowledge, it's the first time that MPMO framework is applied to the prediction of protein structures. IMPMO-DE follows the general framework of MPMO and each population adopts the DE to optimize its corresponding objective. There are two core issues in the MPMO-based methods, which are

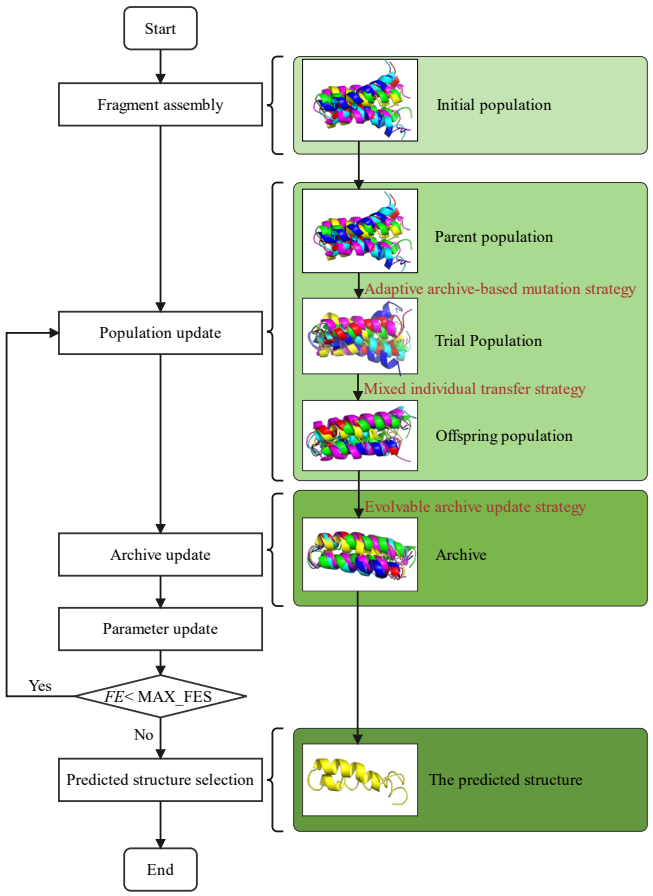


Fig. 3. The flowchart of IMPMO-DE.

population update and archive update. Herein, two novel strategies named AAM and MIT are proposed to enhance the search performance of multiple populations in the population update. Moreover, the EAU strategy is proposed for efficiently archive update.

Fig. 3 describes the flowchart of IMPMO-DE. The initial multiple populations are obtained through the fragment assembly technique. After that, IMPMO-DE goes into a loop to optimize the multiple populations until the number of fitness evaluations is exhausted. Finally, one structure is selected as the predicted structure and is output. The loop of optimization includes three parts, i.e., population update, archive update, and parameter update. 1) For population update, first, trial population is generated according to the AAM strategy and the DE crossover and selection operations. Then, the MIT strategy is executed to transfer these trial individuals to the suitable population and generate the offspring population; 2) For archive update, first, new archived individuals are generated based on random-orient mutation operator and crossover operator. Then, the archive is updated based on the non-dominated sort and crowding distance; 3) For parameter update, the scaling factor F and crossover rate CR of DE are updated.

B. Population Update

1) Adaptive Archive-based Mutation Strategy

IMPMO-DE maintains three populations for three objectives and each population adopts DE to optimize its corresponding objective. During the population update, the

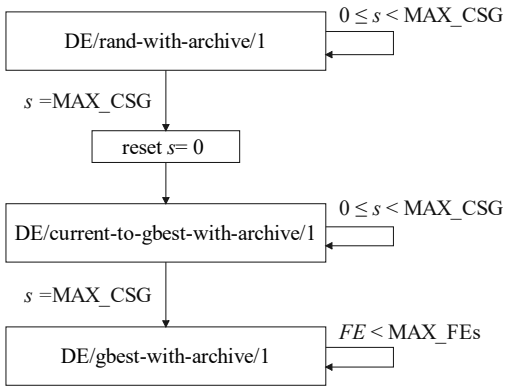


Fig. 4. The switch mechanism in AAM strategy.

mutation operation has an important impact on the exploring ability and also greatly affects the algorithm performance [64], [65]. Therefore, researchers have designed different kinds of mutation operators to improve the search performance. These mutation operators can be roughly divided into two categories: one is random-orient mutation represented by "DE/rand/1" and the other is greedy-orient mutation represented by "DE/best/1". The "DE/rand/1" mutation operator has the ability to efficiently explore the search space, which can maintain the population diversity and is likely to find more promising solutions. The "DE/best/1" mutation operator has strong exploitation ability, which can speed up the convergence and improve the precision of the found solutions. In recent years, some DE algorithms use adaptive operator selection strategy to select different mutation operators in different evolutionary stages and have shown promising performance [66]-[68]. Based on the aforementioned considerations, the AAM strategy is proposed. The AAM strategy includes three novel archive-based mutation operators and an adaptive switch mechanism.

One part of the AAM strategy is the three archive-based mutation operators, which are defined as

1) DE/rand-with-archive/1

$$\boldsymbol{v}_{m,i} = \boldsymbol{x}_{m,r_1} + F \cdot (\boldsymbol{x}_{Ar} - \boldsymbol{x}_{m,r_2}), \tag{8}$$

2) DE/current-to-gbest-with-archive/1

$$\boldsymbol{v}_{m,i} = \boldsymbol{x}_{m,i} + F \cdot (\boldsymbol{Gbest}_m - \boldsymbol{x}_{m,i}) + F \cdot (\boldsymbol{x}_{Ar} - \boldsymbol{x}_{m,r_1}), \tag{9}$$

3) DE/gbest-with-archive/1

$$\boldsymbol{v}_{m,i} = \boldsymbol{Gbest}_m + F \cdot (\boldsymbol{x}_{Ar} - \boldsymbol{x}_{m,r_1}), \tag{10}$$

where $\boldsymbol{x}_{Ar}$ is an individual randomly selected from the archive (the details of the archive are presented in Section III-C). $\boldsymbol{x}_{m,i}$ and $\boldsymbol{Gbest}_m$ represent the i-th individual and best individual in the m-th population respectively. $\boldsymbol{x}_{m,r1}$ and $\boldsymbol{x}_{m,r2}$ are two individuals randomly selected from the m-th population. It's worth noting that r_1 should not be equal to r_2. $\boldsymbol{v}_{m,i}$ is a mutant individual generated through the mutation operator.

The other part of the AAM strategy is the adaptive switch mechanism, which is used to switch the three archive-based mutation operators. The switch mechanism is inspired by the scale-adaptive fitness evaluation [69] and the stagnation-based switch strategy [70]. The idea of scale-adaptive fitness evaluation is to locate the promising regions using the low-accuracy scale evaluation method and improve the precision of solutions through the high-accuracy scale evaluation method. The stagnation-based switch strategy is to control the switch among different fitness evaluation methods. For solving the multi-objective PSP problem, a mutation operator with strong exploration ability (e.g., DE/rand/1 and DE/rand-with-archive/1) is befitting in the early evolutionary stage, while a mutation operator with strong exploitation ability (e.g., DE/best/1 and DE/gbest-with-archive/1) is preferred in the late evolutionary stage. Moreover, mutation operator with balanced exploration and exploitation abilities (e.g., DE/current-to-gbest-with-archive/1) can be used for transition in the middle evolutionary stage.

Algorithm 1 Adaptive Archive-based Mutation (*P, state*)

// Note that $P = \{P_1, P_2, P_3\}$;
Begin
1. **For** m = 1 to 3 **Do**
2. $P_m' = \emptyset$;
3. **For Each** individual $\boldsymbol{x}_{m,i} \in P_m$
4. Randomly select an individual from the archive;
5. Randomly select two individuals from P_m;
6. **If** *state* is equal to 1 **Then**
7. Generate $\boldsymbol{v}_{m,i}$ via Eq. (8);
8. **Else if** *state* is equal to 2 **Then**
9. Generate $\boldsymbol{v}_{m,i}$ via Eq. (9);
10. **Else if** *state* is equal to 3 **Then**
11. Generate $\boldsymbol{v}_{m,i}$ via Eq. (10);
12. **End if**
13. $P_m' = P_m' \cup \{\boldsymbol{v}_{m,i}\}$;
14. **End for**
15. **End for**
16. $P' = \{P_1', P_2', P_3'\}$;
17. **Return** P';
End

Therefore, the switch mechanism in AAM strategy is shown in Fig. 4 and the detailed descriptions are as follows. IMPMO-DE is initialized to use the mutation operator "DE/rand-with-archive/1" with strong exploration ability, in order to locate more promising regions in the early evolutionary stage. It's switched to the mutation operator "DE/current-to-gbest-with-archive/1" if the consecutive stagnation generations s exceeds the predefined limit MAX_CSG. Similarly, it's switched from the "DE/current-to-gbest-with-archive/1" to the "DE/gbest-with-archive/1" if s exceeds the predefined limit again, and "DE/gbest-with-archive/1" would be adopted for the late evolutionary stage. It's worth noting that the switch is one-way because that more exploitations are needed to improve the precision of solutions with the generation increases. The calculation of consecutive stagnation generations is defined as

$$s = \begin{cases} s+1, & \text{if } \forall m \ f_m(\boldsymbol{Gbest}_{m,g+1}) = f_m(\boldsymbol{Gbest}_{m,g}) \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where s is the value of consecutive stagnation generations. It is increased by 1 if no better solutions are found for any objective. Otherwise, s is reset to 0. The pseudo-code of AAM strategy is given in **Algorithm 1**.

Based on the mutant individuals, the binary crossover operator is used to generate the trial individuals, defined as

$$u_{m,i,j} = \begin{cases} v_{m,i,j}, & \text{if } r \le CR \text{ or } j = j_{rand} \\ x_{m,i,j}, & \text{otherwise,} \end{cases} \tag{12}$$

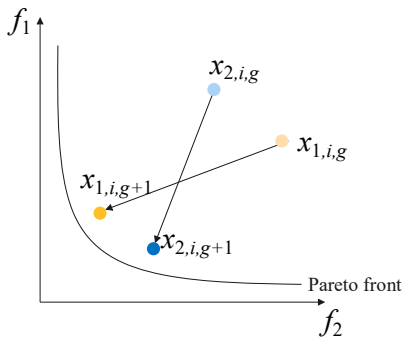


Fig. 5. Illustration of MIT strategy.

where $u_{m,i,j}$ is the j-th dimension of the i-th individual, and r is a random number uniformly generated in range [0, 1]. CR is the crossover rate. j_{rand} is a random integer ranges in [1, D].

Moreover, the selection operator is conducted to compare $\boldsymbol{u}_{m,i}$ with $\boldsymbol{x}_{m,i}$, and the better one is saved. The selection operator for a minimization problem is defined as

$$\boldsymbol{x}_{m,i} = \begin{cases} \boldsymbol{u}_{m,i}, & \text{if } f_m(\boldsymbol{u}_{m,i}) \le f_m(\boldsymbol{x}_{m,i}) \\ \boldsymbol{x}_{m,i}, & \text{otherwise.} \end{cases} \tag{13}$$

Therefore, new individuals are generated through the mutation operators, crossover operator, and selection operator.

2) Mixed Individual Transfer Strategy

IMPMO-DE maintains M populations, and the individuals in each population optimize the corresponding objective. However, an individual that is obtained with the guidance of another objective may even perform better on this objective when comparing with the individual that is obtained with the guidance of this corresponding objective. This is mainly because that EC is a kind of search technique with randomness. For example, as Fig. 5 shows, for a two-objective optimization problem, in the g-th generation, the i-th individual in the population to optimize the objective f_1, i.e., $\boldsymbol{x}_{1,i,g}$, generates a new individual for the next generation, i.e., $\boldsymbol{x}_{1,i,g+1}$. Similarly, the i-th individual in the population to optimize the objective f_2, i.e., $\boldsymbol{x}_{2,i,g}$, generates a new individual for the next generation, i.e., $\boldsymbol{x}_{2,i,g+1}$. Although the two populations perform their duties to optimize their corresponding objectives, i.e., $f_1(\boldsymbol{x}_{1,i,g+1}) \le f_1(\boldsymbol{x}_{1,i,g})$, $f_2(\boldsymbol{x}_{2,i,g+1}) \le f_2(\boldsymbol{x}_{2,i,g})$, better solutions may also be obtained for these two populations through exchanging $\boldsymbol{x}_{1,i,g+1}$ and $\boldsymbol{x}_{2,i,g+1}$, because that $f_1(\boldsymbol{x}_{2,i,g+1}) \le f_1(\boldsymbol{x}_{1,i,g+1})$, $f_2(\boldsymbol{x}_{1,i,g+1}) \le f_2(\boldsymbol{x}_{2,i,g+1})$.

Therefore, the MIT strategy is proposed and described as follows. The MIT strategy mixes all the individuals in all the populations together and then transfers these individuals among the populations according to their performance in each objective. This can reallocate the individuals among the multiple populations to share search information of different populations to accelerate the convergence speed. The pseudo-code of MIT strategy is given in **Algorithm 2** and is described as follows. Firstly, the MIT mixes all the individuals of all the population together in a mixed set. Then, for the m-th population, all the individuals in the mixed set are sorted from better to worse based on the m-th objective. After the sorting, the first $|P_m|$ individuals are regarded as performing well on the m-th objective, and are transferred from the mixed set to the m-th population, where $|P_m|$ is the size of the m-th population. Note that as these $|P_m|$ individuals have transferred from the mixed set to the m-th population, they have been removed from the mixed set (see Line 6 of **Algorithm 2**) and will not be sorted when dealing with the (m+1)-th population.

Algorithm 2 Mixed Individual Transfer (P)

// Note that $P = \{P_1, P_2, P_3\}$;
Begin
1. $SP = P_1 \cup P_2 \cup P_3$; //$SP$ preserves all the mixed individuals
2. **For** m = 1 to 3 **Do**
3. SP = sort all the individuals in SP based on f_m from better to worse;
4. **For** i = 1 to $|P_m|$ **Do**
5. $\boldsymbol{x}_{m,i} = SP_i$; //Transfer this individual to population P_m
6. Delete SP_i from SP;
7. **End for**
8. $P_m = \{\boldsymbol{x}_{m,1}, \boldsymbol{x}_{m,2}, \ldots, \boldsymbol{x}_{m,|Pm|}\}$;
9. **End for**
10. $P = \{P_1, P_2, P_3\}$;
11. **Return** P;

End

The MIT strategy can fasten the convergence speed of multiple populations through transferring the individuals to more suitable populations. Moreover, if an individual is transferred from one population to another population, its search direction is changed in the decision space, and the diversity of the Pareto set is improved to some extent.

C. Evolvable Archive Update

The first proposed MPMO-based method, i.e., CMPSO used archive to store the Pareto solutions and new individuals are only generated in the multiple populations. However, the individuals in the archive may be underutilized. In fact, these archived individuals are also evolvable. Therefore, more promising solutions can be generated based on these archived individuals through evolution by using suitable mutation operator and crossover operator.

The novel EAU strategy is proposed in this paper. The EAU strategy generates more promising solutions through random-orient mutation operator and crossover operator. The random-orient mutation operator is defined as

$$\boldsymbol{a}_i^{'} = \boldsymbol{a}_i + F_i \cdot (\boldsymbol{a}_{r_1} - \boldsymbol{a}_{r_2}), \tag{14}$$

where $\boldsymbol{a}_i$ is the i-th individual in the archive and $\boldsymbol{a}_i$' is the corresponding mutant individual. $\boldsymbol{a}_{r1}$ and $\boldsymbol{a}_{r2}$ are two different individuals randomly selected from the archive.

The crossover operator is defined as

$$a_{i,j}^{''} = \begin{cases} a_{i,j}^{'}, & \text{if } r \le CR_i \text{ or } j = j_{rand} \\ a_{i,j}, & \text{otherwise.} \end{cases} \tag{15}$$

The scaling factor F_i and crossover rate CR_i for the i-th individual in the archive are set as

$$F_i = rand_i[0,1], \tag{16}$$

$$CR_i = rand_i[0,1]. \tag{17}$$

All the new evolved individuals are merged into the archive. However, this will result in that not all the individuals in the archive are non-dominated or the size of the archive excess the archive size. Therefore, the EAU strategy also adopts the non-dominated ranking method [31] on the merged archive to only preserve the non-dominated individuals. Moreover, if the number of the preserved non-dominated individuals is still larger than the archive size, the crowding distance method [71]

Algorithm 3 Evolvable Archive Update (*A, A'*)

Begin
1. $A = A \cup P$;
2. A = Non-dominated solutions of A;
3. **For Each** $\boldsymbol{a} \in A$ **Do**
4. Randomly select two individuals from archive A;
5. Generate F and CR via Eqs. (16) and (17);
6. Generate mutant individual $\boldsymbol{a}'$ via Eq. (14);
7. Generate new archived individual $\boldsymbol{a}''$ based on $\boldsymbol{a}'$ via Eq. (15);
8. Calculate the energy function values of $\boldsymbol{a}''$;
9. $A' = A' \cup \{\boldsymbol{a}''\}$;
10. **End for**
11. $A = A \cup A'$;
12. A = Non-dominated solutions of A;
13. **If** $|A|$ > Predefined Archive Size **Then**
14. Delete some individuals from A based on crowding distance;
15. **End if**
16. **Return** A;

End

is further used to delete some crowding individuals. The pseudo-code of evolvable archive update is given in **Algorithm 3**.

D. Parameter Update

IMPMO-DE maintains multiple populations, with each population adopting the DE to optimize the corresponding objective. It's generally accepted that the scaling factor F and crossover rate CR dramatically affect the performance of DE. Unsuitable setting of these two parameters may result in the stagnation or premature convergence of DE. Therefore, the parameter self-adaptive strategy used in JADE [72] is also used in IMPMO-DE.

For the i-th individual in the m-th population, its scaling factor is generated through the Cauchy distribution, defined as

$$F_{m,i} = randc(\mu_{F,m}, 0.1). \tag{18}$$

The variance of Cauchy distribution is set as 0.1 and the mean is updated in every generation as.

$$\mu_{F,m} = (1-c) \times \mu_{F,m} + c \times mean_L(S_{F,m}), \tag{19}$$

where c is set as 0.1. $S_{F,m}$ is the set containing all the scaling factors that help the trial individuals defeat the parent individuals for the m-th population. $mean_L(\cdot)$ refers to the Lehmer mean defined as

$$mean_L(S_{F,m}) = \frac{\sum_{i=1}^{|S_{F,m}|} \left(F_{m,i}\right)^2}{\sum_{i=1}^{|S_{F,m}|} F_{m,i}}. \tag{20}$$

Besides, the crossover rate is obtained through the Gaussian distribution and is described as

$$CR_{m,i} = randn(\mu_{CR,m}, 0.1). \tag{21}$$

The variance of Gaussian distribution is set as 0.1 and the mean is updated in every generation as

$$\mu_{CR,m} = (1-c) \times \mu_{CR,m} + c \times mean_A(S_{CR,m}), \tag{22}$$

where c is also equal to 0.1. $S_{CR,m}$ is the set that contains the crossover rates of the successfully updated individuals. $mean_A(\cdot)$ is defined as the arithmetic mean, i.e.,

$$mean_A(S_{CR,m}) = \frac{\sum_{i=1}^{|S_{CR,m}|} CR_{m,i}}{\left|S_{CR,m}\right|}. \tag{23}$$

Algorithm 4 IMPMO-DE

Begin
1. Initialize $A = \emptyset$, $P = \{P_1, P_2, P_3\}$, $FE = 0$, $state = 1$, $s = 0$;
2. Generate P through fragment assembly; //Section II-A
3. Evaluate each individual in P and insert all the individuals into A;
4. $FE = FE + |P_1| + |P_2| + |P_3|$;
5. Find the global best individual $\boldsymbol{Gbest}_m$ of each P_m;
6. A = Non-dominated solutions of A;
7. **While** $FE <$ MAX_FEs **Do**
8. P' = **Adaptive Archive-based Mutation** (P, $state$); //**Algorithm 1**
9. $P' = \{P_1', P_2', P_3'\}$;
10. **For** $m = 1$ to 3 **Do**
11. $P_m^* = P_m$; //P_m^* preserves individuals in P_m for parameter update
12. **For** $i = 1$ to $|P_m|$ **Do**
13. Obtain $\boldsymbol{x}_{m,i}$ from P_m; Obtain $\boldsymbol{v}_{m,i}$ from P_m';
14. Generate $\boldsymbol{u}_{m,i}$ based on $\boldsymbol{x}_{m,i}$ and $\boldsymbol{v}_{m,i}$ via Eq. (12);
15. Calculate the energy function values of $\boldsymbol{u}_{m,i}$;
16. Generate $\boldsymbol{x}_{m,i}$ based on $\boldsymbol{x}_{m,i}$ and $\boldsymbol{u}_{m,i}$ via Eq. (13);
17. **End for**
18. $P_m = \{\boldsymbol{x}_{m,1}, \boldsymbol{x}_{m,2}, \ldots, \boldsymbol{x}_{m,|Pm|}\}$;
19. **End for**
20. $P = \{P_1, P_2, P_3\}$;
21. P = **Mixed Individual Transfer** (P); //**Algorithm 2**
22. $A' = \emptyset$; //A' aims to preserve new archived individuals
23. A = **Evolvable Archive Update** (A, A'); //**Algorithm 3**
24. $FE = FE + |P_1| + |P_2| + |P_3| + |A'|$;
25. **For** $m = 1$ to 3 **Do**
26. $S_{F,m} = \emptyset$, $S_{CR,m} = \emptyset$;
27. **For** $i = 1$ to $|P_m|$ **Do**
28. **If** $f_m(\boldsymbol{u}_{m,i}) < f_m(\boldsymbol{x}_{m,i}^*)$ **Then**
29. $S_{F,m} = S_{F,m} \cup \{F_{m,i}\}$;
30. $S_{CR,m} = S_{CR,m} \cup \{CR_{m,i}\}$;
31. **End if**
32. **End for**
33. **If** $S_{F,m} \neq \emptyset$ and/or $S_{CR,m} \neq \emptyset$ **Then**
34. Update $\boldsymbol{\mu}_{F,m}$ and $\boldsymbol{\mu}_{CR,m}$ via Eqs. (19) and/or (22);
35. **End if**
36. **End for**
37. Find the global best individual $\boldsymbol{Gbest}_m$ for each P_m;
38. Update consecutive stagnation generations s via Eq. (11);
39. **If** s is equal to MAX_CSG and $state$ is not equal to 3 **Then**
40. $s = 0$;
41. $state = state + 1$;
42. **End if**
43. **End while**
44. Select one structure as the predicted structure from the archive;

End

E. Predicted Structure Selection

After optimization based on the three energy functions, a set of non-dominated individuals are available in the archive. How to choose one protein structure as the representative one from these candidate solutions is a remained problem to be solved. Many EC-based PSP methods proposed in recent years solve this problem through clustering [24],[25],[29],[30]. Experimental results of these methods verify the effectiveness of the clustering method. The clustering method used in the advanced EC-based PSP method, i.e., MO3, is also adopted in this paper to select one predicted structure from the candidate solution set.

F. Complete IMPMO-DE and Complexity Analysis

The framework of IMPMO-DE mainly contains three parts: 1) population update; 2) archive update; 3) parameter update. The pseudo-code of IMPMO-DE is shown in **Algorithm 4**.

Assume that the total size of multiple populations and the size of archive are N and N_{Ar} respectively, and the number of energy functions is M. The complexity within one generation is analyzed. For population update, it takes $O(N)$ to generate new individuals. Then, it takes $O(MN\log N)$ to sort the individuals and takes $O(N)$ to transfer the individuals. For archive update, it takes $O(N_{Ar})$ for the archive to generate new individuals. Then, it takes $O(MN^2)$ for the non-dominated sort and takes $O(N^2)$ to calculate the crowding distance. For parameter update, it takes $O(N)$ to update F and CR. Therefore, the overall computational complexity is $O(N+MN\log N+N+N_{Ar}+MN^2+N^2+N)$, which is $O(MN^2+N_{Ar})$. Because that N_{Ar} is equal to N in this paper, the complexity can be reduced to $O(MN^2)$.

IV. Experimental Study

The parameter setting of IMPMO-DE is summarized as follows. The total size of multiple populations is set as 60, and the size of each population is 20 because that there are three populations for the three corresponding energy functions. The size of archive A is set as 60. The maximum fitness evaluation number MAX_FEs is set as 18000. The maximum consecutive stagnation generations MAX_CSG is set as 5, and *state* is initialized to 1 to use the mutation operator "DE/rand-with-archive/1". The initial scaling factor F and crossover rate CR for each population are both set as 0.5, and the self-adaptive parameter c is set as 0.1. Besides, Pyrosetta [58] is used to calculate energy function SASA, while the energy functions Rwplus [55] and AACE18 [60] are calculated through the executable files. Links to the executable files are provided in their published paper respectively, i.e., Rwplus (https://zhanggroup.org//RW/) and AACE18 (http://vakser.compbio.ku.edu/main/resources.php). All the experiments are carried out in the same environment, i.e., use Core i7 with 8GB RAM on Ubuntu 20.04 LTS and compile with Python 3.8.10.

The IMPMO-DE is measured by three frequently-used performance metrics and is compared with five advanced EC-based PSP methods on 28 representative proteins. The parameter and component effects are analyzed to find the optimal setting. Moreover, the conflicting relationship among any two objectives is also analyzed to verify the feasibility of using multi-objective optimization algorithms. Last, the IMPMO-DE is tested on all the available TFM proteins up to 404 residues presented in the 14th Critical Assessment of Protein Structure Prediction competition (CASP14) and is compared with all the competitors of CASP14.

A. Performance Metrics

Three different metrics are adopted in this paper to measure the similarity between protein native structure and the predicted structure, i.e., root mean squared error (RMSD) [73], global distance test total score (GDT_TS) [74], and template modeling score (TM-score) [75].

RMSD is the most frequently adopted performance metric for PSP and is defined as

$$\mathrm{RMSD}(a,b)=\sqrt{\frac{1}{n}\sum_{i=1}^{n}|r_{ai}-r_{bi}|^2}, \tag{24}$$

where a and b respectively represent protein native structure and predicted structure after rotation matrix transform through Kabsch algorithm [73]. n is the number of identical atoms between the native structure and predicted structure. r_{ai} and r_{bi} represent the Cartesian coordinates of the i-th atom of the native structure and predicted structure respectively. The unit of RMSD is angstrom (Å).

Besides, another popular performance metric, i.e., GDT_TS, is adopted by the CASP competition and defined as

$$\mathrm{GDT\_TS}=\frac{GDT\_P_1+GDT\_P_2+GDT\_P_4+GDT\_P_8}{4}, \tag{25}$$

where GDT_P_x refers to the proportion of the residue pairs that the distance between the predicted structure and native structure is smaller than xÅ. GDT_TS ranges in [0,100]. The larger the value of GDT_TS, the higher the similarity between the native structure and the predicted structure.

TM-score measures the topology similarity between the native structure and the predicted structure. Compared with traditional performance metric (e.g., RMSD), TM-score owns two advantages: 1) TM-score pays more attention to measuring the overall similarity of structures, rather than local structural changes; 2) TM-score introduces a scale related with the length of amino acid sequence. The distance between residues is normalized, so that the value of TM-score is independent of the length of amino acid sequence. TM-score is formulated as

$$\text{TM-score}=\max\left[\frac{1}{L_{target}}\sum_{i}^{L_{common}}\frac{1}{1+\left(\frac{d_i}{d_0\left(L_{target}\right)}\right)^2}\right], \tag{26}$$

where L_{target} is the number of amino acid residues of the predicted structure, and L_{common} is the same number of residues in the predicted structure and the native structure. Besides, d_i is the distance between the i-th residue of the predicted structure and the i-th residue of the native structure, and d_0 is a predefined parameter to normalize d_i. TM-score ranges in (0, 1]. According to strict statistical analysis of protein databases [76], TM-score less than 0.17 indicates that the predicted structure is not related to the native structure, while TM-score greater than 0.5 indicates that the predicted structure and the native structure generally have the same folding pattern.

B. Comparison with EC-based Approaches

In order to make comparison with EC-based approaches and verify the effectiveness of IMPMO-DE, 28 representative proteins are selected as the test set. These proteins are representative because that they cover three classes of proteins, i.e., $\boldsymbol{\alpha}$ class, $\boldsymbol{\beta}$ class, and $\boldsymbol{\alpha}/\boldsymbol{\beta}$ class. The information of this

TABLE I
THE TEST SET OF 28 REPRESENTATIVE PROTEINS

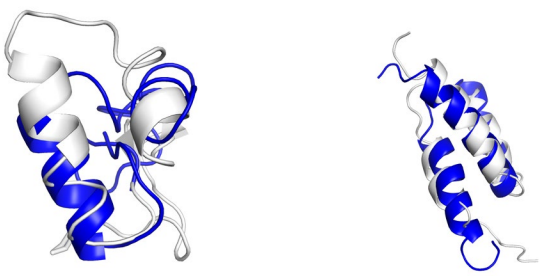
Protein ID	Class	Protein ID	Class	Protein ID	Class	Protein ID	Class
1AB1	$\boldsymbol{\alpha}/\boldsymbol{\beta}$	1F7M	$\boldsymbol{\beta}$	1SXD	$\boldsymbol{\alpha}$	2MR9	$\boldsymbol{\alpha}$
1BDD	$\boldsymbol{\alpha}$	1G26	$\boldsymbol{\beta}$	1UTG	$\boldsymbol{\alpha}$	2P5K	$\boldsymbol{\alpha}/\boldsymbol{\beta}$
1CRN	$\boldsymbol{\alpha}/\boldsymbol{\beta}$	1I6C	$\boldsymbol{\beta}$	1ZDD	$\boldsymbol{\alpha}$	2P6J	$\boldsymbol{\alpha}$
1DFN	$\boldsymbol{\beta}$	1K36	$\boldsymbol{\beta}$	2GB1	$\boldsymbol{\alpha}/\boldsymbol{\beta}$	2P81	$\boldsymbol{\alpha}$
1E0G	$\boldsymbol{\alpha}/\boldsymbol{\beta}$	1MSI	$\boldsymbol{\alpha}/\boldsymbol{\beta}$	2JUC	$\boldsymbol{\alpha}$	2PMR	$\boldsymbol{\alpha}$
1E0M	$\boldsymbol{\beta}$	1Q2K	$\boldsymbol{\alpha}/\boldsymbol{\beta}$	2JZQ	$\boldsymbol{\alpha}/\boldsymbol{\beta}$	3P7K	$\boldsymbol{\alpha}$
1ENH	$\boldsymbol{\alpha}$	1ROP	$\boldsymbol{\alpha}$	2KDL	$\boldsymbol{\alpha}$	3V1A	$\boldsymbol{\alpha}$

TABLE II
THE RMSD AND *P*-VALUE RESULTS BETWEEN IMPMO-DE AND FIVE EC-BASED PSP METHODS

Protein ID	IMPMO-DE	AIMOES	MO3	MOPSO	MODE	MO4
1AB1	6.84	**6.77**	7.52	9.80	7.38	6.82
1BDD	**3.97**	6.95	-	5.64	4.98	4.66
1CRN	6.70	-	**5.56**	7.57	-	6.79
1DFN	**4.53**	7.65	7.45	-	-	8.33
1E0G	**6.77**	7.28	-	-	8.10	8.88
1E0M	**5.55**	5.94	8.00	-	6.49	7.52
1ENH	**3.54**	6.67	11.99	8.92	7.80	4.60
1F7M	**7.41**	9.71	-	-	-	10.95
1G26	8.54	**5.57**	-	-	-	5.89
1I6C	7.30	8.02	-	8.47	7.76	**6.93**
1K36	**7.75**	10.15	-	-	8.34	9.04
1MSI	**9.58**	9.59	-	-	-	10.38
1Q2K	5.78	**4.27**	7.93	-	-	6.53
1ROP	**2.09**	-	3.22	3.51	3.01	2.73
1SXD	**9.09**	12.12	-	-	10.82	12.35
1UTG	**6.02**	-	-	11.13	-	7.91
1ZDD	2.64	4.45	3.26	2.15	2.50	**1.87**
2GB1	**5.76**	6.48	-	-	8.26	7.62
2JUC	**5.92**	7.54	10.80	-	-	7.70
2JZQ	**3.66**	9.62	-	-	-	7.87
2KDL	**6.49**	-	-	10.29	7.72	7.12
2MR9	5.31	7.42	6.68	-	-	**5.19**
2P5K	8.84	**8.52**	9.23	-	-	8.89
2P6J	**3.14**	10.82	6.54	9.44	6.29	6.20
2P81	**4.06**	6.43	4.64	6.28	4.76	4.67
2PMR	**5.98**	6.43	10.12	-	-	7.74
3P7K	2.46	-	3.01	-	-	**1.51**
3V1A	2.24	3.96	**2.23**	-	-	3.72
+(IMPMO-DE is better)		19	14	10	13	22
-(IMPMO-DE is worse)		4	2	1	1	6
p-value		2.76E-04	5.04E-04	9.77E-04	1.22E-04	4.61E-04

test set is presented in TABLE I and more detailed information of this test set is available in TABLE S.I in the Supplementary File. The prediction results include the similarity between the predicted structure and the native structure, which is measured by three performance metrics, i.e., RMSD, GDT_TS, and TM-score. The prediction results of IMPMO-DE on this test set is summarized in TABLE S.II in the Supplementary File.

IMPMO-DE is compared with five advanced EC-based PSP approaches (i.e., AIMOES [28], MO3 [24], MOPSO [29], MODE [30], MO4 [25]) to measure the performance of IMPMO-DE. TABLE II summarizes the RMSD results of IMPMO-DE and five compared methods, with the best results indicated by **boldface**. Moreover, Wilcoxon signed rank test [77] is used to analyze the significant differences between IMPMO-DE and the five compared methods. If the *p* value of Wilcoxon signed rank test is smaller than 0.05, it indicates that



PDB ID: 1AB1
RMSD=6.84Å

PDB ID: 1BDD
RMSD=3.97Å

Fig. 6. The alignment results of protein 1AB1 and 1BDD. The native structures are colored white and the structures predicted by IMPMO-DE are colored blue.

TABLE III
THE *P*-VALUE RESULTS BETWEEN IMPMO-DE AND ITS FIVE VARIANTS WITH DIFFERENT COMPONENTS

IMPMO-DE	IMPMO-DE-1	IMPMO-DE-2	IMPMO-DE-3	IMPMO-DE-no-MIT	IMPMO-DE-no-EAU
+	22	21	20	20	26
-	6	7	8	8	2
p-value	1.37E-04	2.13E-04	2.55E-04	8.71E-04	3.44E-06

IMPMO-DE significantly outperforms the compared method. It's worth noting that the results of the compared methods are obtained directly from their published papers and the results not provided in these published papers are marked with "-". According to the Wilcoxon signed rank test results, IMPMO-DE significantly outperforms all the five advanced EC-based PSP methods, with all the *p*-values smaller than 0.05.

Moreover, in order to show the prediction performance of IMPMO-DE more visually, the predicted structure is aligned with the native structure for each protein. The alignment results of the first two proteins in this test set, i.e., protein 1AB1 and protein 1BDD are presented in Fig. 6. The native structures are colored white and the structures predicted by IMPMO-DE are colored blue. The alignment results of all the proteins are available in Fig. S.I in the Supplementary File.

C. *Component Effects*

IMPMO-DE follows the framework of MPMO, with some improved components are made and integrated. Therefore, these components are worth to discuss and analyze. The components include: 1) The AAM strategy, which may make better balance of exploration and exploitation abilities of IMPMO-DE; 2) The MIT strategy, which may accelerate the convergence speed; and 3) The EAU strategy, which may generate more promising solutions based on archive individuals. TABLE III shows the *p*-value results between IMPMO-DE and its five variants with different components, and the detailed experimental results to calculate the *p*-value are available in TABLE S.III in the Supplementary File. IMPMO-DE-*i* (*i*=1,2,3) represent the IMPMO-DE method with "DE/rand-with-archive/1", "DE/current-to-gbest-with-archive/1", and "DE/gbest-with-archive/1" respectively. IMPMO-DE-no-MIT represents the IMPMO-DE method without MIT strategy. IMPMO-DE-no-EAU represents the IMPMO-DE methods without EAU strategy, i.e., the archive individuals are not updated through the EAU strategy and only the non-dominated sort is carried out in the archive update process. Experimental results show that, IMPMO-DE outperforms all its compared methods, with *p* value smaller than 0.05, which significantly support the effectiveness of each component.

Moreover, the specific roles of these novel strategies are further discussed. The AAM strategy includes three archive-based mutation operators and adaptively switch among these mutation operators to better balance the exploration and exploitation abilities. Compared with the method that use single mutation operator "DE/rand-with-archive/1", IMPMO-DE can convergence faster during the search process. Compared with the method that use single mutation operator "DE/gbest-with-archive/1", IMPMO-DE can obtain better results with smaller RMSD and smaller energy function value after the complete search, because that IMPMO-DE has a

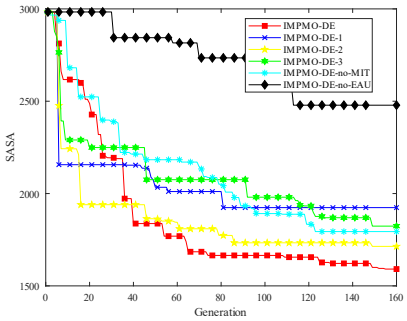


Fig. 7. Convergence curve of protein 1AB1 on energy function SASA.

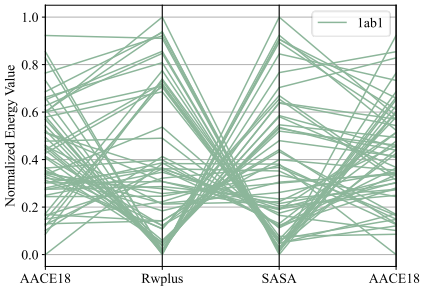


Fig. 8.Parallel coordinate plot of protein 1AB1.

TABLE IV
THE *P*-VALUE RESULTS BETWEEN IMPMO-DE AND ITS THREE VARIANTS WITH DIFFERENT PARAMETER SETTINGS

IMPMO-DE (MAX_CSG=5)	MAX_CSG = 0	MAX_CSG = 3	MAX_CSG = 7
+	22	18	17
-	6	10	11
p-value	1.37E-04	5.68E-02	9.93E-02

smaller probability to fall into the local minimum. Compared with the method that use single mutation operator "DE/current-to-gbest-with-archive/1", IMPMO-DE may have a better balance on exploration and exploitation based on the suitable setting of consecutive stagnation generations *s*. Compared with the method that does not use MIT strategy, IMPMO-DE has a faster convergence speed, especially in the early stage of the search process. Compared with the method that does not use EAU strategy, IMPMO-DE can generate more promising individuals based on the archives and obtain smaller energy function value.

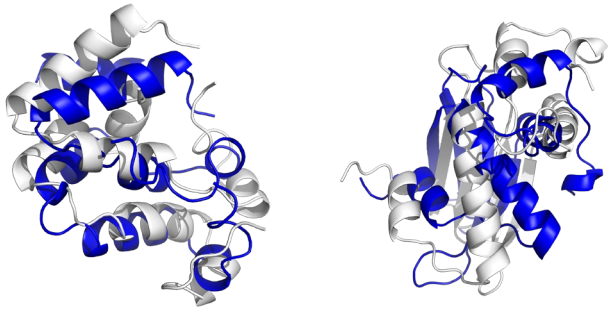
Fig. 7 shows the convergence curve of IMPMO-DE and five variants of protein 1AB1 on the energy function SASA. Similar results can be obtained on all the energy functions for all the proteins and are available in Fig. S.III in the Supplementary File. Experimental results show that, on average, The IMPMO-DE-3 that use "DE/gbest-with-archive/1" convergence fastest in the early stage, while it has a bigger energy function value in the end because that this method is easy to fall into the local minimum; The IMPMO-DE-1 that use "DE/rand-with-archive/1", the IMPMO-DE-no-MIT that does not use MIT strategy, and the IMPMO-DE-no-EAU that does not use EAU strategy, have slow convergence speed and perform worse than IMPMO-DE in the end, because that these three methods have fewer fitness evaluations on refining the solutions; The IMPMO-DE-2 that use "DE/current-to-gbest-with-archive/1" obtain relative good performance, while still worse than the IMPMO-DE. To sum up, these three novel strategies of IMPMO-DE, i.e., the AAM strategy, the MIT strategy, and the EAU strategy, improve the optimizing ability and therefore improve the prediction accuracy of IMPMO-DE.

D. Influence of Parameter Settings

IMPMO-DE has one parameter MAX_CSG that may affect its performance. MAX_CSG means the maximum consecutive stagnation generations, and determines when the mutation operator is switched. A parameter sensitivity analysis is executed to find the best MAX_CSG, and the Wilcoxon signed rank test is used to analyze the significant differences among different parameter settings. MAX_CSG is set as 0, 3, 5, and 7 respectively. When MAX_CSG equals to 0, it means to use mutation operator "DE/rand-with-archive/1" in the whole process. The *p*-value results between IMPMO-DE and its three variants with different parameter settings are presented in TABLE IV, and the detailed experimental results to calculate the *p*-value are available in TABLE S.IV in the Supplementary File. When MAX_CSG is set as 5, the method outperforms the compared method that set MAX_CSG as 0. However, there is no significant difference on *p* value among the settings that MAX_CSG equals to 3, 5, and 7 respectively. IMPMO-DE with MAX_CSG equals to 5 performs slightly better than the other two compared methods, although it's no significant. Therefore, MAX_CSG is set as 5 in this paper.

E. Conflicting Relationship of Objectives

IMPMO simultaneously optimizes three objectives, i.e., SASA, Rwplus, and AACE18. The conflicting relationship between any two objectives should be verified. Therefore, the parallel coordinate plot [78] is adopted to analyze the conflicting relationship. Parallel coordinate plot is frequently used to visualize high-dimensional multivariate data. The parallel coordinate plot contains multiple vertical axes with parallel and equidistant distribution, and different vertical axes represent different normalized objectives. A predicted structure is described by a broken line with vertices on vertical axes. Two objectives are conflict if the line segments between the corresponding and adjacent vertical axes are crossed. Fig. 8 describes the parallel coordinate plots of proteins 1AB1. Similar plots can be obtained for other proteins and are available in Fig. S. IV in the Supplementary File. These plots show that line segments between any two adjacent vertical



Protein ID: T1027
GDT_TS=34.85

Protein ID: T1029
GDT_TS=39.00

Fig. 9.The alignment results of protein T1027 and T1029. The native structures are colored white and the structures predicted by IMPMO-DE are colored blue.

axes are crossed, which means that any two objective functions are conflict with each other.

F. Comparison with CASP Approaches

In order to further verify the effectiveness of the proposed IMPMO-DE, we compare IMPMO-DE with the advanced approaches that participated in the CASP14 competition and test the approaches on all available TFM proteins up to 404 residues. Detailed prediction results of IMPMO-DE on this test set are available in TABLE S.V in the Supplementary File. Herein, to show the prediction performance of IMPMO-DE more visually, the native structures and the predicted structures of the first two proteins in this test set, i.e., protein T1027 and protein T1029, are aligned respectively and presented in Fig. 9. The native structures are colored white and the predicted structures are colored blue. The alignment results of all the proteins are available in Fig. S.II in the Supplementary File.

Moreover, three advanced approaches, i.e., AlphaFold2, Yang-Server (http://yanglab.qd.sdu.edu.cn/index.html), and BAKER-ROSETTASERVER (http://robetta.bakerlab.org/), are selected to make comparison with IMPMO-DE, and the GDT_TS analysis of these four approaches are described in TABLE V. TABLE V shows that IMPMO-DE can generate structures with high accuracy for small-size proteins, such as T1029, T1031, T1033, T1040, T1064, and T1070-D1, and shows competitive performance compared with the Yang-Server and BAKER-ROSETTASERVER. The primary cause is that proteins with small number of residues generally have small conformational search space, which enables IMPMO-DE to generate proteins with high accuracy. However, the prediction accuracy of IMPMO-DE on large-size proteins such as T1037 and T1042 remains limited, which still leaving room for further study.

Moreover, we calculate the average Z-score with threshold of -2.0 to rank IMPMO-DE among all the approaches participated in the CASP14 TFM competition. The Z-score is calculated according to the mean and standard deviation of GDT_TS results, and models with Z-score below the tolerance threshold would be set to -2.0. All the GDT_TS results are available on the CASP14 website (https://predictioncenter.org/casp14/results.cgi?view=targets&model=first&groups_id=&tr_type=all&dm_class=fm). Fig. 10 shows the average Z-score (>-2.0) for IMPMO-DE (colored with red) and all the approaches (colored with white) that participated in CASP14 TFM competition. IMPMO-DE ranks 56th among all the 140 approaches, and the average Z-score (>-2.0) of IMPMO-DE is 0.2094. Three advanced approaches, i.e., AlphaFold2, Yang-Server, and BAKER-ROSETTASERVER, rank 1th, 53th, and 74th, respectively. Therefore, it can be concluded that the proposed IMPMO-DE show competitiveness and can perform better than half of the approaches in this CASP14 TFM test even though it is still worse than some very famous but maybe very complex approaches, e.g., AlphaFold2. In CASP 14, AlphaFold2 leaved other approaches far behind, mainly owing to the design and utilization of end-to-end architecture and attention mechanism.

TABLE V
THE GDT_TS RESULTS BETWEEN IMPMO-DE AND THREE ADVANCED METHODS

Protein ID	Number of residues	IMPMO-DE	Yang-Server	BAKER-ROSETTA SERVER	AlphaFold2
T1027	99	34.85	32.83	35.35	61.11
T1029	125	39.00	40.40	43.60	44.60
T1031	95	49.21	28.16	25.00	87.37
T1033	100	34.50	30.75	46.25	87.50
T1037	404	22.09	52.97	16.46	87.62
T1038-D1	114	31.14	34.65	23.25	89.47
T1039	161	21.43	28.57	39.75	82.30
T1040	130	35.38	23.27	20.19	71.73
T1041	242	23.97	53.93	25.00	90.70
T1042	276	20.76	46.01	14.58	84.51
T1043	148	26.35	11.65	20.78	83.45
T1049	134	49.63	58.95	66.60	93.10
T1064	92	29.35	20.65	26.36	86.96
T1070-D1	76	36.84	28.62	26.32	63.82
T1074-D1	132	32.95	50.95	47.73	89.77
T1090	191	34.03	48.02	46.83	89.02

Although IMPMO-DE cannot achieve high prediction accuracy especially on large-size proteins when comparing with some deep-learning approaches, IMPMO-DE owns two advantages compared with deep-learning approaches: 1) IMPMO-DE is a more general approach because it is independent on protein templates and is suitable to deal with newly discovered proteins without similar known protein structures; and 2) IMPMO-DE is more explainable and straightforward because it simulates the process of biological evolution and folds a protein from one-dimensional amino acid sequence to three-dimensional structure directly. Therefore, we believe that IMPMO-DE is a new artificial intelligence (AI) approach that opens a promising optimization-based evolutionary and explainable way for efficient PSP rather than deep learning approaches like

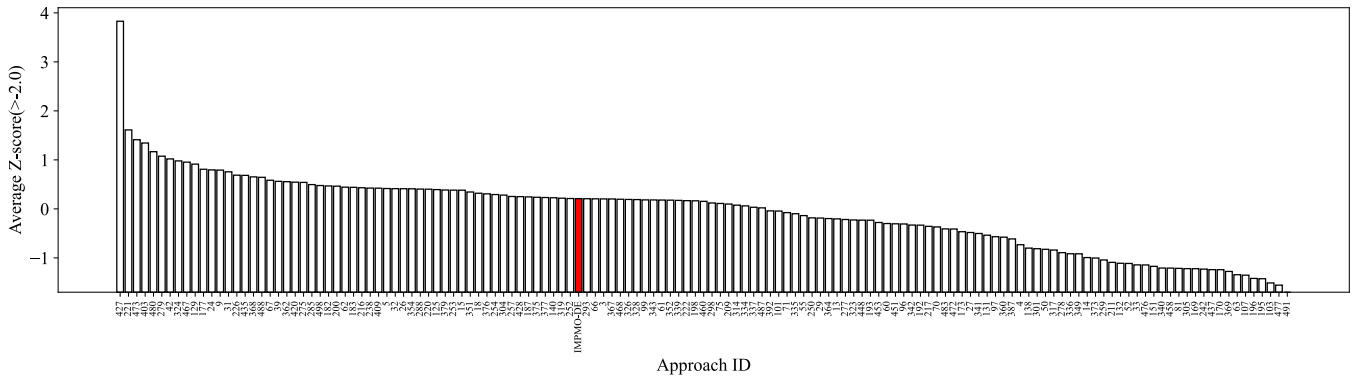


Fig. 10. Average Z-score results with penalty threshold -2.0 for IMPMO-DE (colored with red) and all the approaches (colored with white) that participated in CASP14 TFM competition. Each scale on the horizontal axis represents an approach, whose ID is recorded in CASP14 or is IMPMO-DE.

AlphaFold2. AlphaFold2 is an end-to-end approach that cannot well explain why an input amino acid sequence is corresponding to the predicted structure. However, IMPMO-DE is an evolutionary optimization-based approach that evolves to find the optimal structure, in which whether a structure is good or poor can be well explained by the three explainable objectives. Moreover, the IMPMO-DE may be more promising especially for newly discovered proteins without similar known protein structures in the protein data bank.

V. Conclusion

This paper establishes a novel multi-objective PSP model and proposes the IMPMO-DE algorithm to solve it. To the best of our knowledge, it is the first time that MPMO framework is applied in PSP. In order to enhance the search performance, three major improvements are made to extend the MPMO framework. First, the AAM strategy is designed to better balance the exploration and exploitation of IMPMO-DE in different evolutionary stages. Second, the MIT strategy is proposed to reallocate individuals among different populations to accelerate the convergence speed. Third, the EAU strategy can generate more promising solutions through evolving the archived individuals.

IMPMO-DE is tested on 28 representative proteins and compared with five advanced EC-based approaches. IMPMO-DE is also tested on all available TFM proteins up to 404 residues presented in the CASP14 and compared with the competitors. Experimental results show that IMPMO-DE outperforms the compared advanced EC-based approaches and can perform above average on the CASP14 TFM test set. Parameter sensitivity and component effect are analyzed to find the best setting. The conflicting relationship among three objectives is also discussed.

We will carry out further researches on EC-based PSP in future work, mainly on two aspects. On one aspect, we will try to improve the prediction speed through incorporating parallel and distributed techniques [79]-[81] with EC-based PSP approaches. On the other aspect, we will try to further improve the prediction accuracy by utilizing more biological knowledge related to proteins and combining EC with novel techniques like attention mechanism and/or learning mechanism [82]-[84]. We will also test IMPMO-DE on more proteins, and compare it with more advanced approaches.

References

[1] E. M. Marcotte, M. Pellegrini, H. L. Ng, D. W. Rice, T. O. Yeates, and D. Eisenberg, "Detecting protein function and protein-protein interactions from genome sequences," *Science*, vol. 285, no. 5428, pp. 751-753, 1999.

[2] Z. Q. Li, C. J. Lu, Z. P. Xia, Y. Zhou, and Z. Luo, "X-ray diffraction patterns of graphite and turbostratic carbon," *Carbon*, vol. 45, no. 8, pp. 1686-1695, 2007.

[3] R. Danev, H. Yanagisawa, and M. Kikkawa, "Cryo-electron microscopy methodology: current aspects and future directions," *Trends Biochem. Sci.*, vol. 44, no. 10, pp. 837-848, 2019.

[4] A. Bax, "Multidimensional nuclear magnetic resonance methods for protein studies," *Curr. Opin. Struct. Biol.*, vol. 4, no. 5, pp. 738-744, 1994.

[5] J. Moult, K. Fidelis, A. Kryshtafovych, T. Schwede, and A. Tramontano, "Critical assessment of methods of protein structure prediction (CASP)—Round XII," *Proteins*, vol. 86, no. S1, pp. 7-15, 2018.

[6] D. Xu and Y. Zhang, "Ab initio protein structure assembly using continuous structure fragments and optimized knowledge-based force field," *Proteins*, vol. 80, no. 7, pp. 1715-1735, 2012.

[7] C. A. Rohl, C. E. Strauss, K. M. Misura, and D. Baker, "Protein structure prediction using Rosetta," *Methods Enzymol.*, vol. 383, pp. 66-93, 2004.

[8] J. Jumper *et al.*, "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, no. 7873, pp.583-589, 2021.

[9] M. Baek *et al.*, "Accurate prediction of protein structures and interactions using a three-track neural network," *Science*, vol. 373, no. 6557, pp. 871-876, 2021.

[10] C. B. Anfinsen, "Principles that govern the folding of protein chains," *Science,* vol. 181, no. 4096, pp. 223-30, 1973.

[11] Z. H. Zhan, L. Shi, K. C. Tan, and J. Zhang, "A survey on evolutionary computation for complex continuous optimization," *Artif. Intell. Rev.*, vol. 55, no. 1, pp. 59-110, 2022.

[12] Z. G. Chen, Z. H. Zhan, S. Kwong, and J. Zhang, "Evolutionary computation for intelligent transportation in smart cities: A survey," *IEEE Comput. Intell. Mag.*, vol. 17, no. 2, pp. 83-102, 2022.

[13] J. Y, Li, Z. H. Zhan, and J. Zhang, "Evolutionary computation for expensive optimization: A survey," *Mach. Intell. Res.*, vol. 19, no. 1, pp. 3-23, 2022.

[14] M. Dorn, L. S. Buriol, and L. C. Lamb, "A hybrid genetic algorithm for the 3-D protein structure prediction problem using a path-relinking strategy," in *Proc. IEEE Congr. Evol. Comput.*, 2011, pp. 2709-2716.

[15] X. G. Zhou, C. X. Peng, J. Liu, Y. Zhang, and G. J. Zhang, "Underestimation-assisted global-local cooperative differential evolution and the application to protein structure prediction," *IEEE Trans. Evol. Comput.*, vol. 24, no. 3, pp. 536-550, 2020.

[16] B. Borguesan, M. Barbachan e Silva, B. Grisci, M. Inostroza-Ponta, and M. Dorn, "APL: An angle probability list to improve knowledge-based metaheuristics for the three-dimensional protein structure prediction," *Comput. Biol. Chem.*, vol. 59, pp. 142-157, 2015.

[17] R. Clausen and A. Shehu, "A multiscale hybrid evolutionary algorithm to obtain sample-based representations of multi-basin protein energy landscapes," in *Proc. ACM Conf. Bioinform., Comput. Biol., Health Informatics*, 2014, pp. 269-278.

[18] X. H. Hao, G. J. Zhang, X. G. Zhou, and X. F. Yu, "A novel method using abstract convex underestimation in ab-initio protein structure prediction for guiding search in conformational feature space," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 13, no. 5, pp. 887-900, 2016.

[19] X. Hao, G. Zhang, and X. Zhou, "Conformational space sampling method using multi-subpopulation differential evolution for de novo protein structure prediction," *IEEE Trans. Nanobiosci.*, vol. 16, no. 7, pp. 618-633, 2017.

[20] X. Hao, G. Zhang, and X. Zhou, "Guiding exploration in conformational feature space with Lipschitz underestimation for ab-initio protein structure prediction," *Comput. Biol. Chem.*, vol. 73, pp. 105-119, 2018.

[21] G. Zhang, X. Zhou, X. Yu, X. Hao, and L. Yu, "Enhancing protein conformational space sampling using distance profile-guided differential evolution," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 14, no. 6, pp. 1288-1301, 2017.

[22] V. Cutello, G. Narzisi, and G. Nicosia, "A multi-objective evolutionary approach to the protein structure prediction problem," *J. R. Soc. Interface*, vol. 3, no. 6, pp. 139-151, 2006.

[23] B. R. Brooks *et al.*, "CHARMM: The biomolecular simulation program," *J. Comput. Chem.*, vol. 30, no. 10, pp. 1545-1614, 2009.

[24] S. Gao, S. Song, J. Cheng, Y. Todo, and M. Zhou, "Incorporation of solvent effect into multi-objective evolutionary algorithm for improved protein structure prediction," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 15, no. 4, pp. 1365-1378, 2018.

[25] Z. Lei, S. Gao, Z. Zhang, M. C. Zhou, and J. Cheng, "MO4: A many-objective evolutionary algorithm for protein structure prediction," *IEEE Trans. Evol. Comput.*, vol. 26, no. 3, pp. 417-430, 2021.

[26] B. Olson, K. D. Jong, and A. Shehu, "Off-lattice protein structure prediction with homologous crossover," in *Proc. Genet. Evol. Comput. Conf.*, 2013, pp. 287-294.

[27] B. Olson and A. Shehu, "Multi-objective optimization techniques for conformational sampling in template-free protein structure prediction," in *Intl. Conf. on Bioinf. and Comp. Biol.*, 2014, pp. 143-148.

[28] S. Song, S. Gao, X. Chen, D. Jia, X. Qian, and Y. Todo, "AIMOES: Archive information assisted multi-objective evolutionary strategy for

ab initio protein structure prediction," *Knowledge-Based Syst.*, vol. 146, pp. 58-72, 2018.

[29] S. Song, J. Ji, X. Chen, S. Gao, Z. Tang, and Y. Todo, "Adoption of an improved PSO to explore a compound multi-objective energy function in protein structure prediction," *Appl. Soft. Comput.*, vol. 72, pp. 539-551, 2018.

[30] X. Chen, S. Song, J. Ji, Z. Tang, and Y. Todo, "Incorporating a multiobjective knowledge-based energy function into differential evolution for protein structure prediction," *Inf. Sci.*, vol. 540, pp. 69-88, 2020.

[31] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182-197, 2002.

[32] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712-731, 2007.

[33] Z. H. Zhan, J. Li, J. Cao, J. Zhang, H. S. Chung, and Y. Shi, "Multiple populations for multiple objectives: A coevolutionary technique for solving multiobjective optimization problems," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 445-463, 2013.

[34] J. Wang, W. Zhang, and J. Zhang, "Cooperative differential evolution with multiple populations for multiobjective optimization," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 2848-2861, 2016.

[35] L. Zhang *et al.*, "Cooperative artificial bee colony algorithm with multiple populations for interval multiobjective optimization problems," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 5, pp. 1052-1065, 2018.

[36] Y. R. Naidu and A. K. Ojha, "Solving multiobjective optimization problems using hybrid cooperative invasive weed optimization with multiple populations," *IEEE Trans. Syst. Man, Cybern.*, vol. 48, no. 6, pp. 821-832, 2016.

[37] J. Liang *et al.*, "Utilizing the relationship between unconstrained and constrained pareto fronts for constrained multiobjective optimization," *IEEE Trans. Cybern.*, vol. 53, no. 6, pp. 3873-3886, 2023.

[38] L. M. Antonio and C. A. C. Coello, "Coevolutionary multiobjective evolutionary algorithms: Survey of the state-of-the-art," *IEEE Trans. Evol. Comput.*, vol. 22, no. 6, pp. 851-865, 2018.

[39] Z. G. Chen *et al.*, "Multiobjective cloud workflow scheduling: A multiple populations ant colony system approach," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 2912-2926, 2019.

[40] G. Yao, Y. Ding, Y. Jin, and K. Hao, "Endocrine-based coevolutionary multi-swarm for multi-objective workflow scheduling in a cloud system," *Soft Comput.*, vol. 21, no. 15, pp. 4309-4322, 2017.

[41] X. Zhang, Z. H. Zhan, W. Fang, P. Qian, and J. Zhang, "Multipopulation ant colony system with knowledge-based local searches for multiobjective supply chain configuration," *IEEE Trans. Evol. Comput.*, vol. 26, no. 3, pp. 512-526, 2022.

[42] S. Z. Zhou, Z. H. Zhan, Z. G. Chen, S. Kwong, and J. Zhang, "A multi-objective ant colony system algorithm for airline crew rostering problem with fairness and satisfaction," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 11, pp. 6784-6798, 2021.

[43] S. C. Liu, Z. G. Chen, Z. H. Zhan, S. W. Jeon, S. Kwong, and J. Zhang, "Many-objective job-shop scheduling: A multiple populations for multiple objectives-based genetic algorithm approach," *IEEE Trans. Cybern.*, vol. 53, no. 3, pp. 1460-1474, 2023.

[44] L. J. Wu *et al.*, "Real environment-aware multisource data-associated cold chain logistics scheduling: A multiple population-based multiobjective ant colony system approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 23613-23627, 2022.

[45] J. Y. Li *et al.*, "A multipopulation multiobjective ant colony system considering travel and prevention costs for vehicle routing in COVID-19-like epidemics," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 25062-25076, 2022.

[46] X. F. Liu, Z. H. Zhan, Y. Gao, J. Zhang, S. Kwong and J. Zhang, "Coevolutionary particle swarm optimization with bottleneck objective learning strategy for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 4, pp. 587-602, 2019.

[47] Q. T. Yang, Z. H. Zhan, S. Kwong, and J. Zhang, "Multiple populations for multiple objectives framework with bias sorting for many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 27, no. 5, pp. 1340-1354, 2023.

[48] R. L. J. Dunbrack and F. E. Cohen, "Bayesian statistical analysis of protein side-chain rotamer preferences," *Protein Sci.*, vol. 6, no. 8, pp. 1661–1681, 1997.

[49] D. Gront, D. W. Kulp, R. M. Vernon, C. E. Strauss, and D. Baker, "Generalized fragment picking in Rosetta: design, protocols and applications," *PLoS One*, vol. 6, no. 8, p. e23294, 2011.

[50] Y. Zhang, "Progress and challenges in protein structure prediction," *Curr. Opin. Struct. Biol.*, vol. 18, no. 3, pp. 342-348, 2008.

[51] D. A. Pearlman *et al.*, "AMBER, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structural and energetic properties of molecules," *Comput. Phys. Commun.*, vol. 91, no. 1, pp. 1-41, 1995.

[52] X. Zhu, P. E. M. Lopes, and A. D. MacKerell Jr, "Recent developments and applications of the CHARMM force fields," *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, vol. 2, no. 1, pp.167-185, 2012.

[53] R. Salomon-Ferrer, D. A. Case, R. C. Walker, "An overview of the Amber biomolecular simulation package," *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, vol. 3, no. 2, pp. 198-210, 2013.

[54] L. Wesson and D. Eisenberg, "Atomic solvation parameters applied to molecular dynamics of proteins in solution," *Protein Sci.*, vol. 1, no. 2, pp. 227–235, 1992.

[55] J. Zhang and Y. Zhang, "A novel side-chain orientation dependent potential derived from random-walk reference state for protein fold selection and structure prediction," *PloS One*, vol. 5, no. 10, pp. 1-13, 2010.

[56] M. K. Islam and M. Chetty, "Clustered memetic algorithm with local heuristics for ab initio protein structure prediction," *IEEE Trans. Evol. Comput.*, vol. 17, no. 4, pp. 558-576, 2013.

[57] D. Eisenberg and A. D. McLachlan, "Solvation energy in protein folding and binding," *Nature*, vol. 319, no. 6050, pp. 199–203, 1986.

[58] S. Chaudhury, S. Lyskov, and J. J. Gray, "PyRosetta: a script-based interface for implementing molecular modeling algorithms using Rosetta," *Bioinformatics*, vol. 26, no. 5, pp. 689–691, 2010.

[59] M. J. Sippl, "Calculation of conformational ensembles from potentials of mena force: An approach to the knowledge-based prediction of local structures in globular proteins," *J. Mol. Biol.*, vol. 213, no. 4, pp. 859–883, 1990.

[60] L. Anishchenko, P. J. Kundrotas, and I. A. Vakser, "Contact potential for structure prediction of proteins and protein complexes from Potts model," *Biophys. J.*, vol. 115, no. 5, pp. 809-821, 2018.

[61] C. Zhang, G. Vasmatzis, J. L. Cornette, and C. DeLisi, "Determination of atomic desolvation energies from the structures of crystallized proteins," *J. Mol. Biol.*, vol. 267, no. 3, pp. 707-726, 1997.

[62] S. Miyazawa and R. L. Jernigan, "Estimation of effective interresidue contact energies from protein crystal structures: quasi-chemical approximation," *Macromolecules*, vol. 18, no. 3, pp. 534-552, 1985.

[63] J. Y. Li, Z. H. Zhan, Y. Li, and J. Zhang, "Multiple tasks for multiple objectives: A new multiobjective optimization method via multitask optimization," *IEEE Trans. Evol. Comput.*, 2023, DOI: 10.1109/TEVC.2023.3294307.

[64] M. G. Epitropakis, D. K. Tasoulis, N. G. Pavlidis, V. P. Plagianakos, and M. N. Vrahatis, "Enhancing differential evolution utilizing proximity-based mutation operators," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 99–119, 2011.

[65] W. Gong, Z. Cai, C. X. Ling, and H. Li, "Enhanced differential evolution with adaptive strategies for numerical optimization," *IEEE Trans. Syst. Man, Cybern.*, vol. 41, no. 2, pp. 397–413, 2011.

[66] K. M. Sallam, S. M. Elsayed, R. K. Chakrabortty, and M. J. Ryan, "Improved multi-operator differential evolution algorithm for solving unconstrained problems," in *Proc. IEEE Congr. Evol. Comput.*, 2020, pp. 1–8.

[67] Z. H. Zhan *et al.*, "Cloudde: A heterogeneous differential evolution algorithm and its distributed cloud version," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 3, pp. 704-716, 2017.

[68] Z. H. Zhan, Z. J. Wang, H. Jin, and J. Zhang, "Adaptive distributed differential evolution," *IEEE Trans. Cybern.*, vol. 50, no. 11, pp. 4633-4647, 2020.

[69] S. H. Wu, Z. H. Zhan, and J. Zhang, "SAFE: Scale-adaptive fitness evaluation method for expensive optimization problems," *IEEE Trans. Evol. Comput.*, vol. 25, no. 3, pp. 478-491, 2021.

[70] Y. Q. Wang, J. Y. Li, C. H. Chen, J. Zhang, and Z. H. Zhan. "Scale adaptive fitness evaluation-based particle swarm optimisation for hyperparameter and architecture optimisation in neural networks and deep learning," *CAAI Trans. Intell. Technol.*, vol. 8, no. 3, pp. 849-862, 2023.

[71] C. R. Raquel and P. C. Naval Jr, "An effective use of crowding distance in multiobjective particle swarm optimization," in *Proc. Genet. Evol. Comput. Conf.*, 2005, pp. 257–264.
[72] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, 2009.
[73] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallogr. Sect. A*, vol. 32, no. 5, pp. 922–923, 1976.
[74] A. Zemla, "LGA: A method for finding 3D similarities in protein structures," *Nucleic Acids Res.*, vol. 31, no. 13, pp. 3370–3374, 2003.
[75] Y. Zhang and J. Skolnick, "Scoring function for automated assessment of protein structure template quality," *Proteins*, vol. 57, no. 4, pp. 702–710, 2004.
[76] J. Xu and Y. Zhang, "How significant is a protein structure similarity with TM-score = 0.5?," *Bioinformatics*, vol. 26, no. 7, pp. 889–895, 2010.
[77] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, 2011.
[78] A. Inselberg, "The plane with parallel coordinates," *Vis. Comput.*, vol. 1, no. 2, pp. 69–91, 1985.
[79] J. Y. Li, K. J. Du, Z. H. Zhan, H. Wang, and J. Zhang, "Distributed differential evolution with adaptive resource allocation," *IEEE Trans. Cybern.*, vol. 53, no. 5, pp. 2791-2804, 2023.
[80] X. F. Liu, Z. H. Zhan, and J. Zhang, "Resource-aware distributed differential evolution for training expensive neural-network-based controller in power electronic circuit," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6286-6296, 2022.
[81] Z. H. Zhan *et al.*, "Matrix-based evolutionary computation," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 6, no. 2, pp. 315-328, 2022.
[82] Y. Jiang, Z. H. Zhan, K. C. Tan, and J. Zhang, "Knowledge learning for evolutionary computation," *IEEE Trans. Evol. Comput.*, 2023, DOI: 10.1109/TEVC.2023.3278132.
[83] Z. H. Zhan, J. Y. Li, S. Kwong, and J. Zhang, "Learning-aid evolution for optimization," *IEEE Trans. Evol. Comput.*, vol. 27, no. 6, pp. 1794–1808, 2023.
[84] J. Y. Li, Z. H. Zhan, K. C. Tan, and J. Zhang, "A meta-knowledge transfer-based differential evolution for multitask optimization," *IEEE Trans. Evol. Comput.*, vol. 26, no. 4, pp. 719-734, 2022.

Jun Hong (Student Member, IEEE) received the B.S. degree in computer science and technology from South China University of Technology, Guangzhou, China, in 2022, where he is currently pursuing the Ph.D. degree in computer science and technology with the School of Computer Science and Engineering.

His research interests include multi-objective optimization, many-objective optimization, evolutionary algorithms, neural networks, and their applications in bioinformatics such as protein structure prediction and drug discovery.

Zhi-Hui Zhan (Fellow, IEEE) received the bachelor's and Ph.D. degrees in computer science from Sun Yat-Sen University, Guangzhou China, in 2007 and 2013, respectively.

He is currently a Changjiang Scholar Young Professor and Gifted Professor with the College of Artificial Intelligence, Nankai University, Tianjin, China. His current research interests include evolutionary computation, swarm intelligence, and their applications in real-world problems.

Prof. Zhan is an IEEE Fellow. He was a recipient of the IEEE Computational Intelligence Society (CIS) Outstanding Early Career Award in 2021, the Outstanding Youth Science Foundation from National Natural Science Foundations of China (NSFC) in 2018, and the Wu Wen-Jun Artificial Intelligence Excellent Youth from the Chinese Association for Artificial Intelligence in 2017. He is listed as the World's Top 2% Scientist for both Career-Long Impact and Year Impact in Artificial Intelligence, and is listed as the Elsevier Highly Cited Chinese Researcher in Computer Science. He is currently an Associate Editor of the *IEEE Transactions on Evolutionary Computation*, the *IEEE Transactions on Emerging Topics in Computational Intelligence*, the *IEEE Transactions on Systems, Man and Cybernetics: Systems*, and the *IEEE Transactions on Artificial Intelligence*.

Langchong He received the Bachelor's degree and the Ph. D. degree in Analytical Chemistry from the Northwest University, Xi'an, China, in 1982 and 1998, respectively.

He is currently a professor with the School of Pharmacy, Xi'an Jiaotong University, Xi'an, China. His current research interests include affinity chromatography, instrumental analysis, and drug safety evaluation.

Prof. He established a novel technique for drug discovery and ligand-receptor interaction study. He is one of the Most Cited Chinese Researchers for his exceptional research performance in the field of Pharmacy by Elsevier from 2020 to 2022. He is a Founding Editor and Editor-in-Chief of *Journal of Pharmaceutical Analysis*, the first international English Journal in the pharmaceutical analysis field.

Zongben Xu received the PhD degree in mathematics from Xi'an Jiaotong University (XJTU), in 1987.

He is the Director of the Institute for Information and System Sciences in XJTU. His research interests include intelligent information processing and applied mathematics. He was elected as a member of the Chinese Academy of Science, in 2011. He was a recipient of the National Natural Science Award of China, in 2007 and the Winner of the CSIAM Su Buchin Applied Mathematics Prize, in 2008. He delivered a speech with the International Congress of Mathematicians, 2010. He serves as the chief scientist for the National Basic Research Program of China (973 Project).

Jun Zhang (Fellow, IEEE) obtained his PhD degree in Electrical Engineering from the City University of Hong Kong in 2002.

Prof. Zhang's research contributions span over 500 peer-reviewed publications, of which more than 220 appear in *IEEE Transactions*. His research interests include Computational Intelligence, and Power Electronic Circuits.

Currently, Prof. Zhang serves as an associate editor for both the *IEEE Transactions on Artificial Intelligence* and the *IEEE Transactions on Cybernetics*.