Dean&Francis

ISSN 2959-409X

# NP-Hard Optimization in HP Model Protein Folding: A Systematic Review of Simulated Annealing, Genetic Algorithm, Particle Swarm Optimization, and Tabu Search

**Chenyi Wang**

[1]*College of Arts and Sciences, The University of Case Western Reserve, 24 Thwing Center, Cleveland, OH 44106-1715, United States*
*Corresponding author: cxw818@case.edu*

**Abstract:**

Protein folding prediction in the HP model is an NP-hard problem which highly needs effective heuristic methods to optimize. Protein folding prediction in the HP model is an NP-hard problem which highly needs effective heuristic methods to optimize. This paper reviews and compares four well-known meta-heuristic algorithms, namely Simulated Annealing (SA), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Tabu Search (TS), focused on three main aspects: ability to cross the energy barrier, robustness to initial solutions, and computational cost. The results show that each algorithm has its strength, namely, SA searching exceling in early-stage exploration, GA providing a stable solution with population diversity, PSO efficiently achieving rapid convergence while maintaining moderate robustness, and TS escaping local minima. This study does not aim to identify a single optimal method, but to elucidate the contexts and conditions under which each heuristic approach can be effectively applied to the protein folding problem. The review will become a practical guide for selecting an optimization method for protein structure prediction where NP-hard problems exist.

**Keywords:** Protein Folding Prediction; HP Model; Meta-heuristic Algorithms; Simulated Annealing; Computational Biology

# 1 Introduction

Protein folding and structure prediction has been one of the biggest challenges of computational biology in predicting a protein's stable three-dimensional structures from its amino acid sequences [1]. To predict the protein's structure, contemporary biology generally employs computational tools that build models based on energy functions and search strategies to predict fold paths that specify conformational space or biologically possible configuration [1].

Hydrophobic-polar (HP) Lattice Model is the one of most widely used among the models because of simplicity and operability. It partitions amino acid sequences into two groups of hydrophilic (P) and hydrophobic (H) which are combined to two-dimensional or three-dimensional lattice points. The conformation is obtained by self-avoid walks [2]. Subsequently, the contact number between hydrophobic residues is used to evaluate the energy function, and the quality of the conformations is then determined [2].

However, even when the HP model is simplified, the protein folding problem has been proven to be NP-hard in two-dimensional and three-dimensional lattice spaces: as the sequence length increases, the search space grows exponentially [3]. Traditional exhaustive or exact algorithms often impose extremely large computational pressure on this problem, so heuristic and meta-heuristic algorithms are needed to simplify the calculation process [3].

Among various heuristic methods, this paper focuses on selecting four representative algorithms: Simulated Annealing (SA), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Tabu Search (TS). Among them, SA is a stochastic optimization method that can probabilistically escape from local optimal solutions and approach the global optimal solution [4]; the GA algorithm is based on natural selection and genetic mechanisms, and iteratively selects the global optimal solution [5]; the PSO algorithm finds the global optimal solution by setting particles to share individual extreme values and repeatedly iterating [5]; the TS algorithm uses the taboo table mechanism to escape from local search [5].

However, each algorithm has its limitations. Choosing the appropriate algorithm based on the research object and circumstances is a crucial step in building a prediction model. Therefore, this article will focus on the NP-hard optimization problem in protein prediction models, systematically introduce and compare the performance of SA, GA, PSO, and TS in three aspects: the ability to overcome energy barriers, the robustness of the initial solution, and the computational cost. Through these comparisons, the applicable conditions of different algorithms can be clarified, thereby more accurately constructing protein prediction models.

# 2 Algorithms

## 2.1 Simulated Annealing

Simulated Annealing (SA) is a stochastic optimization method based on the simulation of the physical annealing process. It regulates the acceptance of certain states with increased energy from a high temperature to a low temperature, in order to escape local maximum and approach the global optimum.

When applying SA in the HP model, it is necessary to first embed the H/P chain in the lattice to form a self-avoiding walk conformation; then define the energy function, usually the contact number (or its negative value) between non-covalent adjacent H residues. Then, through perturbation operations such as lattice pulling (pull moves), pivot rotation/reflection, etc., randomly transform the current conformation in the domain to generate candidate states [6].

The Metropolis acceptance criterion is the core of SA: let the energy of the current conformation be $E_c$, the energy of the candidate conformation be $E_n$, and let the energy difference be \DeltaE $= E_n - E_c$, If \DeltaE $\leqslant 0$, always accept it; If \DeltaE $\geqslant 0$, accept it with a probability of $exp(-\Delta E / T)$, where T is the current temperature parameter [7]. This mechanism enables the algorithm to allow "bad" movements in the initial high-temperature stage, thus having the opportunity to cross the energy barrier.

The cooling schedule (cooling schedule or annealing schedule) determines the way the temperature T decreases with iterations or time and has a significant impact on the performance of SA. Common cooling methods include exponential( $T_k + 1 = \alpha T_k$ ), linear descent, or logarithmic descent. In the HP model, different cooling schedules have significantly different effects on the efficiency of finding the optimal or near-optimal solution and the ability to cross higher energy barriers [6]. In addition, to reduce the negative impact of SA output on the sensitivity of the initial solution, the commonly used variants include: multiple restarts (restarting), that is, repeating SA with multiple different initial solutions; saving the lowest energy state throughout the SA process rather than just the final state; and running multiple SA instances in parallel to utilize parallel computing resources to improve the robustness of the solution [6].

## 2.2 Genetic Algorithm

Genetic Algorithm (GA) is a meta-heuristic method based on the principles of natural selection and genetics. It optimizes the candidate solutions in the population through parallel search. In the HP model for protein folding, the typical process of GA is to first establish an initial population and conduct fitness evaluation, selection, then perform crossover and mutation on individuals, select the best solution of the current generation and iterate [8]. Its advantages lie in maintaining population diversity and balancing global exploration and local exploitation through the combination of genetic operators, thus approaching low-energy solutions in the vast conformational space [8].

In terms of individual representation, GA usually uses relative encoding of self-avoiding walk (SAW) or absolute lattice coordinates to describe protein conformations. The fitness function is centered on the energy function, and a common definition is to count the number of non-covalent contacts between H-H residues, with a lower value indicating a more stable conformation [8]. Due to the requirement of connectivity and self-avoidance in the HP model, GA often introduces a penalty function mechanism to assign lower fitness to infeasible solutions, guiding the algorithm to converge to physically reasonable conformations [8].

When choosing the selection operator, sorting selection is usually used to ensure the genetic advantage of excellent individuals. When choosing the crossover operator, multi-point crossover is usually selected because it can more effectively recombine the local motifs of different individuals and avoid generating large-scale invalid conformations [9]. When choosing the mutation operator, perturbation is introduced through local rotation, flipping or fragment rearrangement of the chain to enhance population diversity [9].

Recent research has also proposed Hybrid Genetic Algorithm (HGA): by combining GA with local search methods (such as tabu search or hill climbing algorithm), to improve the efficiency of search [8]. Additionally, Parallel Genetic Algorithm (PGA) has emerged: by using multiple subpopulations and migration mechanisms to accelerate convergence and leveraging high-performance computing platforms to accelerate protein instances [8]. These improvements have significantly enhanced the applicability of the GA algorithm.

However, GA also has certain limitations. Compared with simulated annealing, GA is more sensitive to parameter settings and requires more precise parameter tuning. However, it has an advantage in maintaining diversity and avoiding local optima. Overall, GA is suitable for use in environments that require search breadth and stability, especially in scenarios with complex energy landscapes and multiple local maxima in protein folding problems [8].

## 2.3 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a heuristic search algorithm based on the concept of a group. Particle Swarm Optimization (PSO) is a heuristic search algorithm based on the concept of a group. It guides the particle swarm towards the optimal value by continuously updating the particle's velocity and position. It guides the particle swarm towards the optimal value by continuously updating the particle's velocity and position. The particle swarm algorithm consists of particle position and velocity, the best value of the particle (pBest), the best value of the group (gBest), and refresh rules. It is typically suitable for handling continuous or discrete encoded problems.

The particle swarm algorithm consists of particle position and velocity, the best value of the particle (pBest), the best value of the group (gBest), and refresh rules. It is typically suitable for handling continuous or discrete encoded problems.

When applying PSO to protein structure prediction in the HP model, it is necessary to make discrete/adaptive modifications to the standard PSO. When applying PSO to protein structure prediction in the HP model, it is necessary to make discrete/adaptive modifications to the standard PSO. Specifically, particles are usually represented as the conformation of an H/P sequence at a grid point (using relative steps or grid coordinates), and the velocity is defined as the direction from the current conformation to the pBest or gBest conformation. Specifically, particles are usually represented as the conformation of an H/P sequence at a grid point (using relative steps or grid coordinates), and the velocity is defined as the direction from the current conformation to the pBest or gBest conformation. The update operation may include swapping residue positions, local pulling move or neighbor changes to approximately simulate the position update mechanism [10].

The update operation may include swapping residue positions, local pulling move or neighbor changes to approximately simulate the position update mechanism [10].

To avoid premature convergence and local minimum problems, many studies in recent years have combined PSO with other strategies. For example, incorporating tabu search (TS) to introduce historical memory to limit particle backtracking, or establishing subgroups within the group to maintain diversity; and using mutatioTo avoid premature convergence and local minimum problems, many studies in recent years have combined PSO with other strategies.n or pulling strategies to enhance exploration ability [11].

The cooling schedule (or temperature mechanism) is not essentially dependent on temperature like SA, but there are similar control parameters in PSO, such as inertia weight, learning factors (cognitive and social components), and velocity limits. The settin for example, incorporating tabu search (TS) to introduce historical memory to limit particle backtracking, or establishing subgroug of these parameters directly affects the ability to cross energy barriers and the convergence speed.ps within the group to maintain diversity; and using mutation or pulling strategies to enhance exploration ability [11].

Choosing a larger inertia weight and random perturbation helps escape local minimum values, but may lead to a decrease in stability; conversely, a strong contraction learning factor helps to refine the search, but weakens the exploration ability [10].The cooling schedule (or temperature mechanism) is not essentially dependent on temperature like SA, but there are similar control parameters in PSO, such as inertia weight, learning factors (cognitive and social components), and velocity limits. The setting of these parameters directly affects the ability to cross energy barriers and the convergence speed. Choosing a larger inertia weight and random perturbation help escape local minimum values but may lead to a decrease in stability; conversely, a strong contraction learning factor helps to refine the search but weakens the exploration ability [10].

### 2.4 Tabu Search

Tabu Search (TS) is a meta-heuristic method based on local search. It avoids the algorithm from getting stuck in cycles or repeatedly visiting already explored solutions by maintaining a “tabu list” during the search process. The core idea of TS is to expand the search range by using the memory mechanism while allowing a certain degree of non-improving moves, thereby helping the algorithm overcome barriers and escape from local extremums [12].

In the HP model, TS usually starts with a self-avoiding walk (SAW) configuration as the initial solution and defines neighborhood operations (such as local pulling, chain segment rotation, pivot movement), to gradually generate new candidate conformations. In each iteration, the optimal candidate solution in the neighborhood is selected as the next state, even if it may lead to an energy increase, if the configuration is not restricted by the tabu list [12]. The tabu list records the solutions or operations that have been accessed in the recent few steps, thus avoiding the algorithm from reverting to the same energy valley.

Recent improvement methods have introduced adaptive tabu lengths and dynamic memory structures. For example, the tabu length can be adjusted according to the diversity of the current search state or the convergence speed, achieving a more reasonable balance between exploration and exploitation [13]. Additionally, some studies have combined TS with genetic algorithms, using the local search ability of TS to assist the convergence of the global algorithm [13].

In terms of the ability to cross energy barriers, TS effectively avoids getting stuck in local minima through the tabu list and the strategy of allowing non-improving solutions. However, its performance highly depends on the quality of the neighborhood design: a too small neighborhood limits the search range, while a too large neighborhood leads to a rapid increase in computational cost [12]. In terms of the robustness of the initial solution, TS usually has a low dependence on the initial solution, as the taboo mechanism helps the search to jump out of the local area and cover a wider space. In terms of computational cost, the main burden of TS lies in neighborhood generation and tabu list maintenance, but compared to GA and PSO, its population size is small, thus showing lower resource consumption under certain conditions [13].

## 3 Result

This section mainly discusses the evaluation of four meta-heuristic algorithms (SA/Metropolis, GA, PSO, TS) on the HP model. The evaluation dimensions include the ability to Barrier-crossing ability, the Robustness of the initial solution, and the Computational cost. It highlights the relative advantages and limitations of different algorithms in each dimension.

### 3.1 Barrier-crossing ability

Simulated Annealing (SA) allows non-improved solutions during the high-temperature stage based on the Metropolis acceptance criteria, thus demonstrating strong ability to overcome energy barriers when the energy landscape is rugged. However, as the temperature decreases, its exploration ability gradually becomes limited, and thus the cooling scheduling design directly determines its final performance [14].

Genetic Algorithm (GA) recombines the information of different individuals through crossover and mutation operations, enabling it to escape from local optima to some extent, but its ability to overcome energy barriers depends on the design of operator diversity. Multi-point crossover and structure fragment-based mutation perform better than single-point crossover in complex instances [15].

Particle Swarm Optimization (PSO) shares global and individual optimal information at the group level. If the parameters are set reasonably (such as a higher inertia weight), it can effectively explore and overcome energy barriers in the early stage. However, if the group converges too early, its ability to overcome barriers will rapidly decline [16].

Tabu Search (TS) relies on a tabu table to avoid backtracking and local cycling, thus demonstrating significant advantages in overcoming energy barriers. By allowing a certain degree of non-improved solutions and combining dynamic tabu length, TS can often continuously escape from local traps in complex protein instances with an energy landscape [17].

### 3.2 Robustness of the Initial Solution

The robustness of SA is usually low because its search process is closely related to the initial solution. Different initial conformations may lead to different final energy values [14]. The robustness of GA depends on the population size and the method for maintaining population diversity. Larger populations and appropriate elite retention strategies can reduce the influence of the initial solution on the global search and improve computational stability [15]. PSO shows moderate robustness if it can generate a sufficiently dispersed particle swarm during the initialization stage. However, if the initialization is too concentrated, the group may quickly fall into a certain local region, thereby reducing robustness [16].TS has the least dependence on the initial solution because its tabu mechanism ensures that the search process will continuously escape from local regions, thereby largely canceling the influence of the initial solution. This is particularly prominent in high-dimensional HP model instances [17].

### 3.3 Computational Cost

In terms of computational cost, the expense of SA mainly comes from the long cooling process, especially when a slow cooling scheduling is adopted, the time complexity will significantly increase [14].

The computational cost of GA is closely related to the population size and the number of iterations. Large populations can improve the search quality while also bringing higher computational costs [15].

The cost of PSO is mainly related to the number of particles and the number of iterations. Its expense is usually between GA and SA, but evaluating the energy function of all particles in high-dimensional space can lead to a rapid increase in cost [16].

The cost of TS mainly lies in its neighborhood generation and tabu table maintenance. It does not depend on the population size, so overall consumption is not significant. However, if the neighborhood is too large, the computational cost of TS will also increase [17].

## 4 Discussion

Although this paper compared the application of four meta-heuristic algorithms (SA/Metropolis, GA, PSO, and TS) in the HP model, it still has certain limitations. Firstly, the paper selected four classic algorithms for discussion, but with the development of algorithms and computing hardware in recent years, many superior emerging methods (such as quantum-inspired algorithms, deep reinforcement learning) have been incorporated into the use of prediction models [18].

Secondly, although the HP model is of great significance in theoretical research, its highly simplified nature (only distinguishing between H and P types of residues) makes it difficult to fully reflect the complex interactions of real proteins, such as hydrogen bonds, electrostatic interactions, and solvation effects. This leads to a possible gap between the performance of the algorithms in the HP model and the actual protein folding [19]. Moreover, with the breakthroughs in deep learning, the HP model has been replaced by more advanced prediction models, such as the AlphaFold end-to-end neural network model, which can directly predict protein three-dimensional structures close to experimental accuracy by learning from a large-scale structure database [20].

In future research directions, hybrid optimization methods are considered as an important trend in solving the NP-hard problem of protein folding. For example, combining the global exploration ability of GA or PSO with the local development ability of TS can improve the search efficiency and the stability of the solution [21]. At the same time, the combination of deep learning and meta-heuristic methods is also gradually emerging, such as using neural networks to predict initial solutions or energy landscape features and then combining heuristic search to accelerate convergence [22].

## 5 Conclusion

Regarding the protein folding problem, which is an NP-hard issue in the HP model, this paper conducts a comparative analysis of four classic meta-heuristic algorithms: Simulated Annealing (SA/Metropolis), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Tabu Search (TS). Based on the existing research results, the paper compares and discusses these four algorithms from three aspects: the ability to overcome energy barriers, the stability of the initial solution, and the computational cost. The results show that each of the four algorithms has its own advantages and disadvantages.

SA/Metropolis, with the help of temperature control and probability acceptance mechanism, demonstrates strong ability to overcome energy barriers in the early stage. However, it relies on cooling scheduling and has insufficient exploration ability after convergence.

GA maintains good robustness through group evolution and diversity and is suitable for exploring complex energy landscapes. However, its computational performance depends on operator design and parameter tuning, and the computational cost also increases with the increase in population size.

PSO shares the global optimal solution at the group level and can converge quickly and overcome energy barriers but is prone to premature convergence. Its robustness depends on initialization and parameter configuration.

TS avoids local traps by relying on the tabu table and performs well in terms of stability and ability to overcome energy barriers. However, maintaining the tabu table brings higher computational costs.

Overall, by clearly identifying the applicable environmental conditions of different algorithms, this paper provides a reference for researchers to reasonably select optimization strategies in the protein structure prediction task.

## References

1. P. Carracedo-Reboredo, S. Liñares-Blanco, A. Rodríguez-Fernández, J. Cedrón-Cabo, H. Novoa, C. Carballal, et al., A review on machine learning approaches and trends in drug discovery. Briefings in Bioinformatics, vol. 22, no. 6, pp. 1–19 (2021). https://doi.org/10.1093/bib/bbab159

2. C.-H. Yang, Y.-S. Wu, and W.-C. Yeh, Protein folding prediction in the HP model using ions motion optimization with a greedy algorithm. BioData Mining, vol. 11, no. 17, pp. 1–19 (2018). https://doi.org/10.1186/s13040-018-0170-2

3. M. Traykov, K. Traykov, and D. Boiadjiev, Protein folding in 3D lattice HP model using heuristic optimization methods. WSEAS Transactions on Circuits and Systems, vol. 17, pp. 192–200 (2018).

4. T. Guilmeau, J. F. Bonnans, and D. Chikhi, Simulated annealing: a review and a new scheme. In: Proceedings of the 2021 IEEE Statistical Signal Processing Workshop (SSP), pp. 31–35 (2021). https://doi.org/10.1109/SSP49050.2021.9513764

5. C. Rajwar, P. K. Gupta, and R. Kumar, An exhaustive review of the metaheuristic algorithms for engineering problems. Mathematics, vol. 11, no. 3, pp. 1–40 (2023). https://doi.org/10.3390/math11030618

6. A. Möbius, Simulated annealing in the hydrophobic-polar (HP) model: experiments on the 3D136 instance including restarts, comparison of final vs. best-visited states, and cooling schedule effects. Journal of Innovative Materials in Extreme Conditions, vol. 5, issue 1, pp. 9–17 (2024).

7. C. Zhang, Comparative Analysis of Simulated Annealing in Protein Folding Prediction Using HP Models. Theoretical and Natural Science, vol. 75, pp. 197–205 (2025).

8. B. Bošković and J. Brest, Genetic algorithm with advanced mechanisms applied to the protein structure prediction in a hydrophobic–polar model and cubic lattice. Applied Soft Computing, vol. 45, pp. 61–70 (2016). https://doi.org/10.1016/j.asoc.2016.04.001

9. S. P. N. Dubey, R. Kumar, and S. K. Singh, A comparative study on single and multiple point crossovers in a genetic algorithm for HP model-based protein structure prediction. Informatics in Medicine Unlocked, vol. 12, pp. 92–100 (2018). https://doi.org/10.1016/j.imu.2018.07.003

10. M. Rezaei, A. Ahmadi-Javid, B. Mahdavi, A novel algorithm based on a modified PSO to predict 3D structure for proteins in HP model using Transfer Learning. Expert Systems with Applications, vol. 211 (2024). https://doi.org/10.1016/j.eswa.2023.121233

11. Y. Shuchun, L. Xianxiang, T. Xue, M. Pang, Protein structure prediction based on particle swarm optimization and tabu search strategy. BMC Bioinformatics, vol. 23, article 352 (2022). https://doi.org/10.1186/s12859-022-04888-4

12. S. Shmygelska and H. H. Hoos, An improved Tabu Search algorithm for the HP protein folding problem. BMC Bioinformatics, vol. 6, no. 30, pp. 1–19 (2015). https://doi.org/10.1186/1471-2105-6-30

13. H. Chen, J. Zhang, and Q. Li, Hybrid tabu search strategies for protein structure prediction in HP model: adaptive tabu length and cooperative heuristics. Journal of Computational Biology, vol. 27, no. 12, pp. 1778–1792 (2020). https://doi.org/10.1089/cmb.2019.0325

14. M. Zaki, A. Elsayed, and A. Kattan, Cooling schedules and performance trade-offs in simulated annealing for protein structure prediction. Journal of Computational Biology, vol. 27, no. 9, pp. 1342–1355 (2020). https://doi.org/10.1089/cmb.2019.0187

15. P. Singh and R. S. Chauhan, Improved genetic operators for hydrophobic–polar protein structure prediction. Journal of Bioinformatics and Computational Biology, vol. 19, no. 6, pp. 2150027 (2021). https://doi.org/10.1142/S0219720021500278

16. L. Miao, Y. Wang, and C. Zhao, Hybrid particle swarm optimization for HP model-based protein folding prediction. IEEE Access, vol. 9, pp. 55021–55033 (2021). https://doi.org/10.1109/ACCESS.2021.3070000

17. J. Wu and Q. Zhang, Dynamic tabu search strategies for complex protein folding landscapes. BMC Bioinformatics, vol. 22, no. 315, pp. 1–15 (2021). https://doi.org/10.1186/s12859-021-04289-1

18. L. Jumper, R. Evans, A. Pritzel, et al., Highly accurate protein structure prediction with AlphaFold. Nature, vol. 596, no. 7873, pp. 583–589 (2021). https://doi.org/10.1038/s41586-021-03819-2

19. P. Crescenzi, D. Goldman, C. Papadimitriou, et al., Protein structure is hard: complexity results for folding the HP model.

Journal of Computational Biology, vol. 27, no. 12, pp. 1678–1690 (2020). https://doi.org/10.1089/cmb.2019.0331

20. M. Mirjalili, S. Saremi, H. Faris, and S. Mirjalili, Advances in metaheuristic optimization for protein structure prediction: opportunities and challenges. IEEE Access, vol. 8, pp. 149159–149180 (2020). https://doi.org/10.1109/ACCESS.2020.3016654

21. Z. Yuan, L. Zhang, and H. Liu, Hybrid metaheuristic algorithms for complex optimization: a review and perspectives. Information Sciences, vol. 581, pp. 401–426 (2021). https://doi.org/10.1016/j.ins.2021.09.001

22. H. Senior, R. Evans, J. Jumper, et al., Improved protein structure prediction using potentials from deep learning. Nature, vol. 577, no. 7792, pp. 706–710 (2020). https://doi.org/10.1038/s41586-019-1923-7