Journal of Information Security Research

DLINE JOURNALS

Print ISSN: 0976-4143
Online ISSN: 0976-4151

JISR 2025: 16 (1)
https://doi.org/10.6025/jisr/2025/16/1/17-24

Application of Swarm Intelligence Algorithms in Higher Vocational Teaching

Qin Li
Xinxiang Vocational and Technical College
Xinxiang, Henan, 453006, China
liqin73@126.com

ABSTRACT

College Chinese," as a core course in literary majors, faces challenges in achieving its intended goals due to the limitations of traditional teacher-centred teaching methods and students' lack of enthusiasm. This study adopts genetic algorithms, particle swarm algorithms, hybrid frog-leaping methods, and swarm intelligence optimization algorithms to improve this situation. Through practical application, it is found that these methods outperform traditional manual and composite methods. However, in-depth research reveals that swarm intelligence algorithms have lower solving efficiency. To better meet practical application needs, a series of convergence and precision adjustments are applied to the results obtained by the swarm intelligence algorithms. Furthermore, improvements are made to the hybrid frog-leaping algorithm to cater to various engineering improvement requirements. Using artificial intelligence algorithms to guide classroom activities expands students' thinking and inspires their minds to better cope with multiple complex engineering challenges.

Keywords: Swarm Intelligence Algorithms, College Chinese, Teaching Reform

Received: 2 September 2024, Revised 1 December 2024, Accepted 13 December 2024

1. Introduction

With the advent of the 2.0 era, our higher education system is continuously improving, from the foundation of hardware infrastructure and technological applications to the exploration of "new ways of classroom teaching relying on IT," "development of education services relying on the internet," and "exploration of new ways of higher education management in the information age." These advancements are continually driving our higher education progress [1]. With the vigorous development of technology, the central government actively promotes information construction to meet the demand for informatization in today's society [2]. Therefore, researching and developing cutting-edge technologies such as artificial intelligence and big data to improve and optimize the training environment in language classrooms have become crucial tasks in today's society [3]. Through in-depth investigation and analysis, we can better recognize the importance of intelligent classrooms today [4]. We will focus on how to use IT to improve the learning effectiveness of college Chinese and attempt to create more effective training methods [5]. According to a feasible model, artificial intelligence learning data analysis involves multiple steps, such as data collection, preprocessing, storage, analysis, execution, and final results. When achieving the best results, machines will conduct real tests, corrections, and feedback based on a large amount of learning data to provide practical, customized classroom content and other auxiliary resources [6]. Constructing a good database and perfect algorithm model is essential to achieve efficient artificial intelligence. By collecting massive amounts of information and analyzing it, our machines can more accurately identify users' needs and implement personalized learning plans [7].

2. Related Work

With the continuous development of artificial intelligence technology, optimization algorithms have been widely applied in various fields of industry and agriculture. For example, Samuel et al. used the hybrid frog-leaping algorithm to optimize automated micro-irrigation pipes' installation and pipe size, effectively reducing project costs [8]. Rajamohana et al. optimized the main evaporation of a secondary heater using the particle swarm algorithm, enabling the rapid optimization of the PID controller parameters of the controller [9]. Rajamohana et al. used the genetic algorithm to optimize the camshaft assembly process, thereby improving the efficiency and quality of the factory [10]. Introducing the ant-lion algorithm can effectively enhance wind power clusters' energy storage capacity configuration, demonstrating exemplary performance in practical operations [11]. Therefore, we can see that swarm intelligence algorithms have tremendous potential value in multiple fields. Teaching college Chinese courses can bring more fun to our learning, broaden our thinking, and stimulate our creativity, thereby better achieving our course objectives. Therefore, we propose applying swarm intelligence algorithms to the teaching process of this course, enabling them to intelligently assist us in completing the entire teaching process [12,13]. We can also use these algorithms to evaluate the performance of modern artificial intelligence algorithms and guide our classmates in finding better teaching models. This way, we can help develop our creativity, broaden our horizons, and better prepare ourselves for future engineering challenges [14].

3. Teaching Design Based on Swarm Intelligence Algorithms

3.1 Particle Swarm Algorithm

By using position vector *xi* and velocity vector *vi*, we can transform a particle '*I*' without mass into a form with different parameters, where *xi* = [*x1, x2, ..., xn*] represents the number of particles in the population, and *D* represents the number of decision variables. Each particle will continuously adjust its velocity and position during the evolutionary process according to the rules defined by Equation (1).

$$\boldsymbol{x}_i(t+1) = \boldsymbol{x}_i(t) + \boldsymbol{v}_i(t+1) \tag{1}$$

'*t*' refers to the number of iterations, o e" o indicates the inertia weight coefficient, C e" o indicates the acceleration coefficient, and r refers to a random number in the range [0, 1]. They all exhibit a balanced distribution. *pBest* refers to the individual best solution of the *i-th* particle, and *gBest* refers to the global best solution of the entire swarm. According to the velocity change rule defined by Equation (1), the first step is to obtain the ratio of the current velocity to the inertia force, which means that particles can achieve inertia movement based on their current velocity; the second step is introspection, which helps particles understand their past development; the third step is social interaction, which allows particles establish effective communication and collaboration among themselves [15]. For complex multi-objective particle swarm optimization, although there are many effective approaches to choose from, such as guiding particles towards the Pareto front and maintaining the diversity of obtained solutions, dealing with complex situations still poses a challenge due to the lack of global evaluation. After in-depth research, we have established a comprehensive swarm framework to integrate improvement solutions and extract efficient strategies effectively. Based on its effective optimization mechanism, the algorithm is divided into multiple sub-modules to facilitate effective integration between different sub-modules. By studying the relationships between various modules in depth, we can more accurately determine the best arrangement of modules, making the algorithm construction more efficient.

3.2 Genetic Algorithm

Because the genetic algorithm is based on the principles of evolution and genetics, it requires the integration of various knowledge, such as the information about evolution and genetics provided in Figure 1, to achieve effective search.

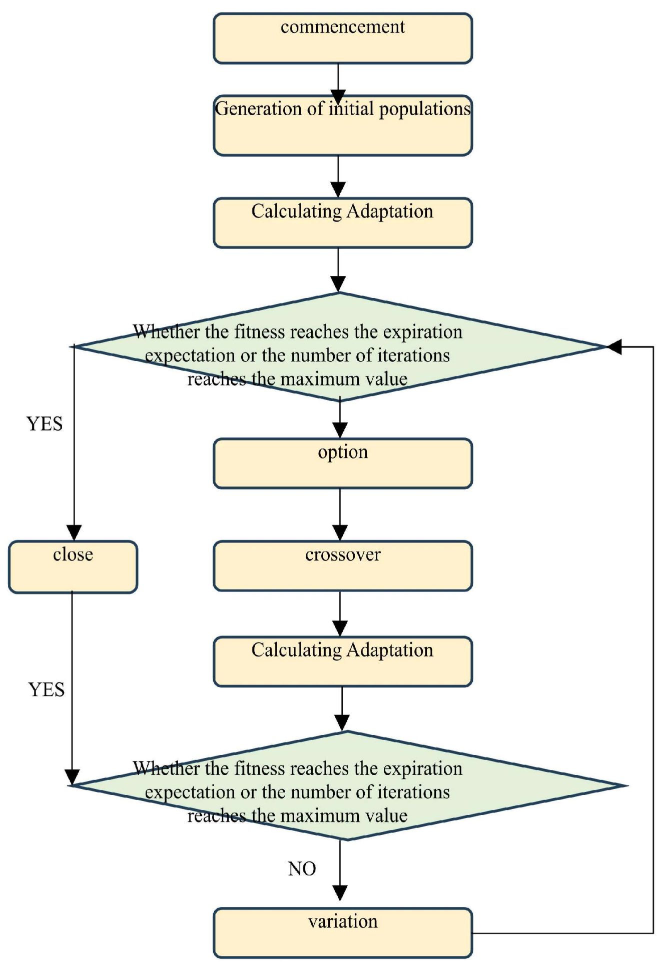

Figure 1. Genetic Algorithm Process Diagram

Although using this type of fitness function is easy to understand, they also have two drawbacks: firstly, they cannot meet the non-negative probability requirement for traditional roulette wheel selection; secondly, sometimes, due to the range of parameter values and differences between features, the calculated average fitness cannot reflect the overall performance of the entire system, thereby reducing the effectiveness of the algorithm. If the objective function is some extremum value, then this parameter is infinite.

$$fit(f(x)) = \frac{1}{1+c+f(x)} \qquad (2)$$

3.3 Hybrid Frog-leaping Algorithm

The hybrid frog-leaping algorithm utilizes a diverse virtual environment to effectively manage the population, where the population information can be widely distributed, enabling better collaboration and achieving self-regulation and self-protection more efficiently. In a swamp, a small collection of frogs is scattered around, shuttling between the swamps, seeking the nearest food sources. These individuals form several cultural genotypes according to a series of rules, and the most outstanding individuals undergo continuous improvement. With the development of time, information exchange and sharing among individuals have become crucial, and the strengths and weaknesses of individuals will have significant impacts on their development. Therefore, individual development depends not only on individual strengths but also on commonalities among individuals, which may play a crucial role in enabling individuals to better fulfill their tasks and improve the quality of cultural genotypes. By blending multiple cultural genotypes, the best information among them can be effectively disseminated, resulting in a stable new population, and optimal results can be achieved by reassigning cultural genotypes, such as operations. Thus, the metaheuristic framework of the hybrid frog-leaping algorithm can be summarized as the evolution of cultural genotypes among individuals and the global mixing and propagation of information among the population. The hybrid frog-leaping algorithm's key lies in global and local search. (1) Firstly, determine the size of the total sample $S = mm$, where m represents the number of cultural genotypes, and n represents the number of frogs in each cultural genotype. (2) Generate a virtual population: For population S within the feasible domain "RR, there are virtual frogs U1. The number of cultural features of the i^{th} frog is represented by U, which can be obtained by decomposing them into "*l2*" and "*ls*," where "u" represents the number of cultural genes carried by the individual, and "u" represents the individual's fitness value $f(j)$. (3) Based on different environmental conditions, group all frog individuals in the population, starting from 1P, and group them from low-to-high to obtain the best results. (4) By sorting S, we can divide this frog population into m cultural genotypes, $Y1$, $Y2$, ..., and Ym, including n frogs. The specific distribution method can be referred to as

$$Y^k = \left[U^k(j), f^k(j)\right| U^k \qquad (3)$$

When m=3, the 1st, 2nd, and 3rd individual frogs will be sequentially arranged in the 1st, 2nd, and 3rd cultural genetic entities, and each individual will have a corresponding relationship. Subsequently, the 4th, 5th, and 6th frogs will also be allocated to the 1st, 2nd, and 3rd cultural, genetic entities in order in this loop. (5) The evolution of cultural genetic entities can be achieved through local search, where y, k respectively represent 1, 2, ... and m. (6) Hybrid recombination operation: Carry out hybrid recombination operation on all cultural, genetic entities at the end of evolution, reconstruct the population, calculate individual fitness values, and rank to determine the global optimal solution replacement. (7) When an individual reaches the optimal position and after a certain number of iterations, the algorithm will stop running, indicating that it has found the optimal solution and, therefore, the optimization is over. If this condition is not met, you need to re-enter the 4th step to continue optimization. Otherwise, the solution ends.

4. Experimental Design and Result Analysis

4.1 Experimental Design

By comparing XUHM's MOPSO and MOEA/DW6 with Coello's MOPSO160, this study found that the population size of XUHM is 100, the archive upper limit for ZDTN is 100, for all other test functions it is 300, while the precision of MOPSO has reached the level of 1.0e-6. Using the *Tchebycheff* method to deal with MOEA/D, we set the population size to 100 for ZDTN and 300 for the other test functions; the number of iterations to WFG7 and WFG9 and 1000 for other test functions; the number of neighbors to 20, the cross-index to 20, the mutation index to 20, the cross-ratio to 1, the mutation ratio to 1/24. In Coello, the size of the set is set to 100; in ZDTN, the size of the archive is set to 100; for other measurement functions, it is set to 300; in ZDTN, it is set to 1000; in WFG7, it is set to 6000; in WFG9, it is set to 9000; for other measurement functions, it is set to 3000; and the mutation ratio is set to 0.5. In addition, this model uses an adaptive grid, dividing each objective into 30 parts. In 30 single attempts, we will evaluate all measurement functions. Typically, classic swarm intelligence algorithms do not need to be developed by professional computer personnel or learners themselves; they can be easily accessed through MATLAB software and can build reasonable fitness functions through the analysis of specific parameters. These parameters can help swarm intelligence algorithms effectively identify specific parameters and effectively guide the development of swarm intelligence algorithms. Generally, we can use some variables to adjust the objective function to obtain a more reasonable fitness function. Although the results of the objective function may not be too complex, they have many different constraints so that we can refer to the "penalty function" in the book "Mathematical Modeling" for design. In this case, we used external reference values as penalty functions.

4.2 Result Analysis

For each function and algorithm, there are 30 operation results, each of which is equivalent to a surplus value. We select the operation result corresponding to the median surplus value as the representative solution set of the algorithm on this function. For the same function, different algorithms draw the distribution diagram of the representative solution set in the objective space into a graph, which facilitates visual comparison. Blue "*" represents the XUHM solution set; green "+" represents the MOEA\D solution *set; red* ":" represents the Coello solution set. For ZDTN, DTLZN, the Coello solution set is far inferior to the XUHM and MOEAID solution sets, and drawing the three groups of solution sets into a graph will make the XUHM and MOEAID solution sets uncomparable. Therefore, two graphs are drawn, the first graph integrates the XUHM and MOEAID solution sets, and the second graph integrates the XUHM and Coello solution sets. The graph is shown in Figure 2. For DTLZN, the MOEAID solution set is concentrated around (1, 0.0), so it cannot be seen in the first graph set. XUHM shows that ZDTN, DTLZN, WFG1, WFG2, and WFG3 perform significantly better than other algorithms. However, from WFG4 to WFG9, XUHM shows little difference compared to other algorithms.

We use the mean and standard deviation to record the distribution of performance evaluation indicators, and the best average values of the algorithms in the example are displayed in bold in the table. As shown in Figure 2, the analysis of the large volume needs to be combined with the generation gap because the increase of some large volumes is due to the rise of the generation gap. The frontier of WFG2 is a straight line, so the overcapacity value is invalid. The experiment shows that only on DTLZN does XUHM present better diversity than MOEAID, which is due to the uneven characteristics of DTLZN causing the *Tchebycheff* decomposition method used by

MOEAID to fail. The results also show that XUHM has more diversity than Coello for each function. Through seven different standard benchmark test functions, we studied and evaluated five different artificial intelligence algorithms, including ASF-BA, EGWO, QDEDA, IWOA, and CWG-GOA, and obtained their averages and standard deviations through 30 single independent runs, as shown in Figure 3. The advantages of ASF-BA are apparent: it can handle F1F4 and FF1 problems more excellently, and its standard deviation has also been significantly improved. In addition, it uses distributed search and population sampling distribution, greatly improving the search process, and uses special thresholds and golden section techniques to enhance its reliability further.

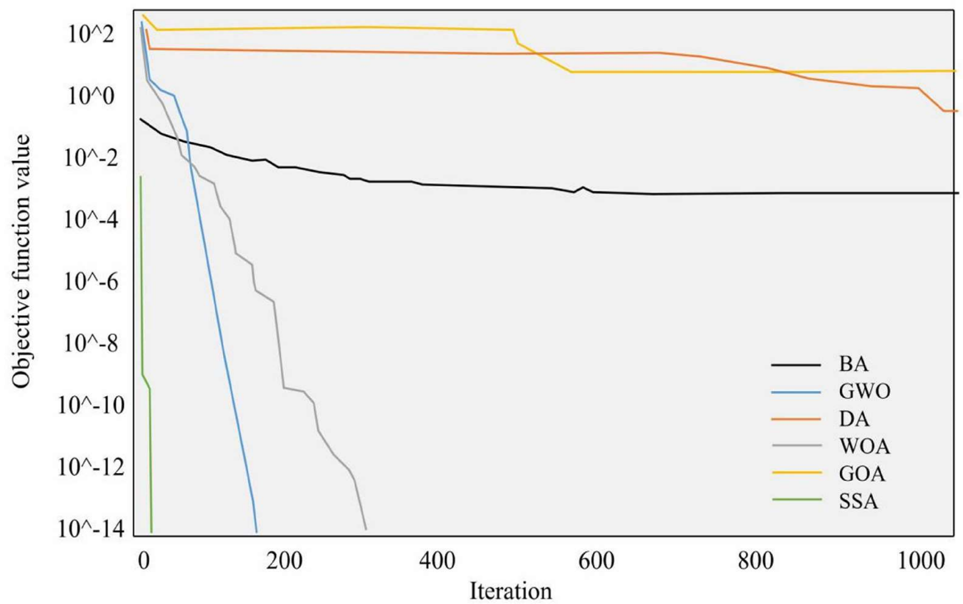

Figure 2. Representation of Solution Sets in Objective Space

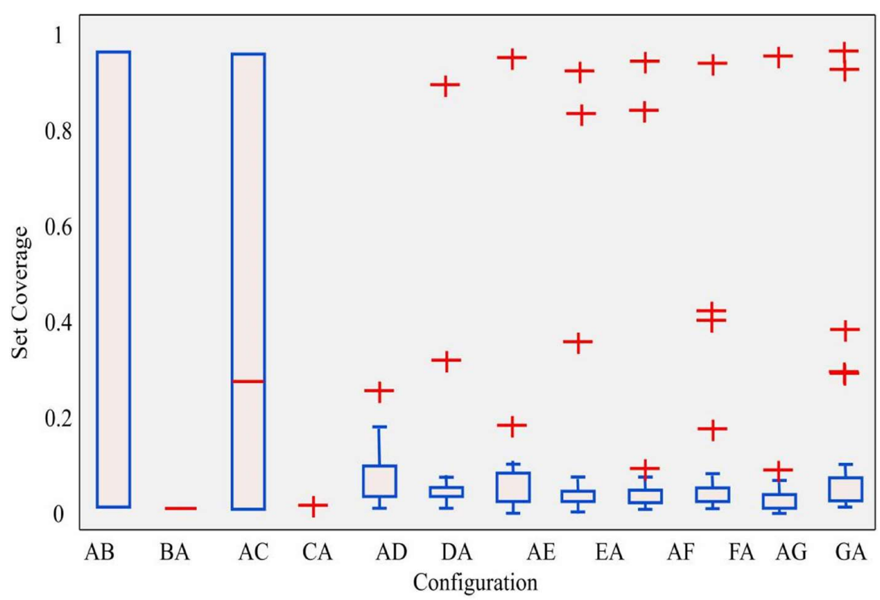

Figure 3. Evaluation of Improvement Results

For ZDTN, random reinitialization leads to excessive randomness, thereby reducing the performance of the algorithm. However, for other functions that require higher randomness, this randomness may be precisely what is needed. Besides these two configurations, the basic configuration A performs the best. Therefore, all strategies are effective. Dimensional leap, optimal substitution, gene exchange, and disregard for individual optimization contribute to convergence; gridization enhances diversity and reduces computational cost; excessive random reinitialization will delay convergence, but this can be acceptable compared to premature convergence.

5. Conclusion

Swarm intelligence algorithms are an artificial intelligence computational technology with broad application prospects. They can effectively help students in the "University Chinese" course better understand optimization theory, expand their thinking horizons, cultivate their innovative capabilities, and solve complex problems. To better realize the fitness of the swarm intelligence algorithm, we use the outlier method to construct a penalty function, and according to the students' ability level, we use the interior point method and mixed method to build a penalty function, to improve the efficiency of the swarm intelligence algorithm. In addition, we have also improved the mixed frog leaping algorithm to enhance its performance. By applying swarm intelligence optimization algorithms to other engineering problems, we can better explore more effective solutions.

References

[1] Juan, D., Hong Wei, Y. (2021). Particle swarm optimization neural network for research on artificial intelligence college English classroom teaching framework. *Journal of Intelligent & Fuzzy Systems, 40*(2), 3655-3667.

[2] Wang, Q. (2021). Application of Clustering Algorithm in Ideological and Political Education in Colleges and Universities. *Journal of Physics: Conference Series, 1852*(3), 032041.

[3] Devos, A. O., Torbenko, I. O., Doroshenko, T. V., et al. (2021). The Application of the Simulation Method in the in Foreign Language Teaching in Higher Education Institutions, the Cognitive Linguistic Approach. *Journal of Educational and Social Research, (4)*, 0072.

[4] Gao, X. (2021). Role of 5G network technology and artificial intelligence for research and reform of English situational teaching in higher vocational colleges. *Journal of Intelligent & Fuzzy Systems, 40*(2), 3643-3654.

[5] Wu, X. (2021). Application of Artificial Intelligence in Higher Vocational English Teaching Mode. *Journal of Physics: Conference Series, 1852*(2), 022089.

[6] Xu, D. (2021). The Application and Analysis of Task-based Language Teaching in English Listening and Speaking Course for English Majors: A Case Study of Sichuan University of Culture and Arts. *Frontier of Higher Education, 3*(4), 10-13.

[7] Wang, S., Yu, T., Li, J., et al. (2022). Application of silicon drift detector in PIXE tomography system of Fudan University. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with*

Materials and Atoms, 512, 108-113.

[8] Samuel, G. G., Rajan, C. C. A. (2015). Hybrid: Particle swarm optimization–genetic algorithm and particle swarm optimization–shuffled frog leaping algorithm for long-term generator maintenance scheduling. *International Journal of Electrical Power & Energy Systems, 65*, 432-442.

[9] Rajamohana, S. P., Umamaheswari, K. (2018). Hybrid approach of improved binary particle swarm optimization and shuffled frog leaping for feature selection. *Computers & Electrical Engineering, 67*, 497-508.

[10] Rajamohana, S. P., Umamaheswari, K. (2017). Hybrid optimization algorithm of improved binary particle swarm optimization (iBPSO) and cuckoo search for review spam detection. *Proceedings of the 9th International Conference on Machine Learning and Computing*, 238-242.

[11] Kaveh, A., Talatahari, S., Khodadadi, N. (2020). Hybrid invasive weed optimization-shuffled frog-leaping algorithm for optimal design of truss structures. *Iranian Journal of Science and Technology, Transactions of Civil Engineering, 44*, 405-420.

[12] Prabakaran, S., SenthilKumar, V., Kavaskar, S. (2017). Hybrid particle swarm optimization algorithm to solve profit based unit commitment problem with emission limitations in deregulated power market. *International Journal of Computer Applications, 167*(5), 37-49.

[13] Zhang, X., Lin, Q., Mao, W., et al. (2021). Hybrid Particle Swarm and Grey Wolf Optimizer and its application to clustering optimization. *Applied Soft Computing, 101*, 107061.

[14] Zeng, X., Nazir, M. S., Khaksar, M., et al. (2021). A day-ahead economic scheduling of microgrids equipped with plug-in hybrid electric vehicles using modified shuffled frog leaping algorithm. *Journal of Energy Storage, 33*, 102021.

[15] Sameh, M. A., Badr, M. A., Badr, M. A. L., et al. (2018). Optimized PIA controller for photovoltaic system using hybrid particle swarm optimization and cuttlefish algorithms. *2018 7th International Conference on Renewable Energy Research and Applications (ICRERA)*. IEEE, 1102-1108.