JTAM (Jurnal Teori dan Aplikasi Matematika)
http://journal.ummat.ac.id/index.php/jtam
p-ISSN 2597-7512 | e-ISSN 2614-1175
Vol. 9, No. 4, October 2025, pp. 1082-1093

Enhancing Adaptive Particle Swarm Optimization Based on Human Social Learning with Human Learning Strategies for the Traveling Salesman Problem

Yusti Qomah[1]*, Bib Paruhum Silalahi[1], Toni Bakhtiar[1]
[1]Department of Mathematics, IPB University, Indonesia
qomahyq25yusti@apps.ipb.ac.id

Article History:
Received : 14-05-2025
Revised : 17-07-2025
Accepted : 18-07-2025
Online : 01-10-2025

Keywords:
Particle Swarm Optimization;
Human Learning Optimization;
Traveling Salesman Problem;
Operations Research.

ABSTRACT

Particle Swarm Optimization (PSO) is a widely used metaheuristic approach for solving optimization problems. Recent developments in this field involve the adaptation of human learning behaviors to enhance algorithmic performance. One such adaptation is the Adaptive Particle Swarm Optimization based on Human Social Learning (APSO-HSL), a variant of PSO that incorporates human-inspired learning strategies. This study aims to enhance the performance of APSO-HSL on the Traveling Salesman Problem (TSP) by incorporating additional human learning strategies. The proposed algorithm, named Modified Adaptive Particle Swarm Optimization–Human Learning Strategies (MAPSO-HLS), integrates learning mechanisms from Human Learning Optimization (HLO), including individual, random, and social learning. This research is classified as applied research and algorithmic experimentation, focusing on the development and modification of a metaheuristic algorithm to solve a well-known combinatorial optimization problem. Benchmark datasets from the Traveling Salesman Problem Library (TSPLIB) are used for evaluation, and all computations and experiments are implemented in Python. The performance of MAPSO-HLS is compared with the original APSO-HSL in terms of shortest distance, convergence rate, and population diversity. A comparison of the shortest distances was conducted using exact solutions and evaluated through percentage deviation. The results show that MAPSO-HLS produces more accurate solutions than APSO-HSL. Convergence analysis reveals that MAPSO-HLS converges faster toward lower objective values. Its advantage is further supported by the diversity analysis, where the diversity curves indicate a better balance between exploration and exploitation.

Crossref
https://doi.org/10.31764/jtam.v9i4.31466

A. INTRODUCTION

Particle Swarm Optimization (PSO) is a meta-heuristic algorithm introduced by Kennedy and Eberhart and is widely used to solve optimization problems. The PSO algorithm is based on natural phenomena, namely the movement of a group of living things, such as birds and fish in finding food (Houssein et al., 2021). Computationally, PSO has advantages in terms of memory usage and speed (Jiyue et al., 2023; Lynn & Suganthan, 2015; Punyakum et al., 2022). The simplicity and efficiency factors have caused PSO to be widely used in various fields and it is considered the most effective method for solving optimization problems (Bangyal et al., 2023). PSO implementations are widely found in various fields, such as health, environment, industry, and commerce (Al-Maamari & Omara, 2015; Ramdhani, 2016).

Despite its strengths, standard PSO often suffers from premature convergence, particularly when parameters are not properly tuned (Guo et al., 2025; Larsen et al., 2016; Ashraf et al., 2022). This leads to limited global exploration, insufficient local exploitation, and reduced solution diversity. To address these limitations, several PSO variants have been proposed, including heterogeneous comprehensive learning PSO (Lynn & Suganthan, 2015), discrete PSO (Zhong et al., 2018), and improved PSO with adaptive initialization techniques (Ashraf et al., 2022), each aiming to enhance convergence and avoid local optima.

A promising direction to overcome these challenges is the integration of human learning behavior into algorithmic design (Roberts-Mahoney et al., 2016; Jarecki et al., 2018; Wang et al., 2017; Du et al. 2022). Human Learning Optimization (HLO) introduces learning strategies inspired by how humans solve problems, through random trial, individual experience, and social imitation. These principles have been shown to improve convergence and solution diversity in complex search spaces (Wang et al., 2014). Building on this, the Adaptive Particle Swarm Optimization based on Human Social Learning (APSO-HSL) algorithm incorporates human social learning into the PSO framework, resulting in improved accuracy, stability, and global search performance (Jiyue et al., 2023). Nonetheless, current implementations of APSO-HSL primarily focus on the social learning component, leaving the potential of individual and random learning strategies underexplored.

Solving combinatorial problems such as the Traveling Salesman Problem (TSP), a classic NP-hard problem requiring the optimal traversal of cities, is a compelling application for metaheuristics (Shaj et al., 2016; Chen et al., 2025). TSP not only represents real-world complexity but also demands a balance between exploration and exploitation due to its exponentially growing solution space (Jedrzejowicz et al., 2024). Although many PSO variants have been applied to TSP, further exploration into the role of adaptive human-inspired learning remains limited.

This study aims to enhance the APSO-HSL algorithm by integrating comprehensive human learning strategies from HLO namely random, individual, and social learning to improve the effectiveness of PSO in solving the TSP. Based on Vahdat et al. (2016), the combination of meta-heuristic algorithms with adaptive learning mechanism has many advantages in solving optimization problems compared to algorithms inspired by natural phenomena. By extending APSO-HSL with a more complete representation of human learning behavior, the proposed approach addresses the gap in existing PSO-based methods that insufficiently balance exploration and exploitation.

B. METHODS

This research is categorized as applied research and algorithmic experimentation, focusing on the development and modification of a metaheuristic optimization to solve a well-known combinatorial optimization problem (TSP). The dataset utilized in this research comprises benchmark instances sourced from the Traveling Salesman Problem Library (TSPLIB). Seven case examples with varying numbers of cities were selected: namely burma14, gr666, gr96, pr226, u574, ulysses16, and ulysses22 (TSPLIB, n.d.). All experiments were conducted using an Intel PC (core i5 @3.09GHz CPU, 4GB RAM).

1. Particle Swarm Optimization (PSO)

The PSO algorithm begins with the particle initialization and initial velocity assignment. The optimal function value and location can be found by using the initial velocity to assess the objective function at each particle location. The particle velocity at that moment, each particle's optimal location, and each particle's optimal neighboring location are then used to determine a new velocity. The particle's location, velocity, and neighbors are updated iteratively until the algorithm reaches the stopping criterion. The new location was determined by adding the old location and modified velocity to maintain the particle within the boundary (Ab Wahab et al., 2015; Jain et al., 2018; Gad, 2022).

To obtain the best solution, each particle moves based on its personal best position (p_{best}) and the global best position (g_{best}) in the swarm. Each particle i will update its velocity v and position x in each iteration $t+1$ using the following equation:

$$v_i^{t+1} = \omega v_i^t + c_1 r_1\left(p_{best_i}^t - x_i^t\right) + c_2 r_2 (g_{best}^t - x_i^t), \tag{1}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}, \tag{2}$$

where i denotes the particle index, t is the current iteration order, ω is the inertia used to balance local exploitation and global exploration, r_1 and r_2 is any number uniformly distributed on [0,1], c_1 and c_2 are the acceleration coefficients, which are two positive constants. Equation (1) is used to calculate the new velocity of the particle based on the distance and last velocity of a position from the particle's best personal experience and the group's best experience. Then, the particle moves towards the new position based on Equation (2) (Piotrowski et al., 2020; Zhong et al., 2018).

2. Adaptive Particle Swarm Optimization based on Social Learning Intelligence (APSO-HSL)

The APSO-HSL algorithm developed by Jiyue et al. (2023) uses a learning strategy based on the multiswarm technique which the diversity of each particle is determined by the division of the swarm and the size of each swarm. Human social learning intelligence is used to adaptively divide the swarm and determine the size of each subswarm. The multiswarm technique in APSO-HSL has the following mathematical definition:

$$\begin{gathered} f_1 = f_{max}, \\ f_2 = f_{\min} + \alpha_1 (f_{max} - f_{\min}), \\ f_3 = f_{\min} + \alpha_2 (f_{max} - f_{\min}), \\ f_4 = f_{\min} + \alpha_3 (f_{max} - f_{\min}), \\ f_5 = f_{\min}, \\ \Omega_j^{\mathrm{t}} = \{x_i^t \mid f_j \leq \text{fitness } (x_i^t) < f_{j+1}, i = 1, 2, \dots, N; j = 1, 2, 3, 4\}. \end{gathered} \tag{3}$$

$f_{\min}$ and f_{max} are the minimum and maximum values of the fitness function. $\{f_k, k = 1, 2, \dots, 5\}$ denotes the subswarm boundaries. Ω_j^{t} denotes the jth subswarm at the tth iteration. x_i^t is the position of the ith particle at the tth iteration. The upper and lower bounds for each particle's fitness are determined using α_k with $k = 1, 2, 3$, where $\alpha_1 = 0.25, \alpha_2 = 0.50, \alpha_3 = 0.75$. The determination of the upper and lower bounds were determined to avoid a large size imbalance of the subswarms.

Every particle was split into three groups according to their learning capacity, each of which had a distinct purpose based on how well they evolved with each iteration. The first group is made up of the swarm's best particles overall ($Gbest$). The second is each subswarm's best particle ($Sbest$). The remaining particles with average fitness, referred to as ordinary particles, make up the third group. The following is a definition of the inertia load operators for the three groups:

$$\omega_i^t = \begin{cases} \dfrac{\dfrac{\sum_{j=1}^{N(t)-1} fitness\left(Sbest_j^t\right)}{(N(t)-1)}}{fitness(x_i^t)}, & if\ x_i^t = Gbest^t \\ \dfrac{fitness\left(Sbest_{j+1}^t\right)}{fitness(x_i^t)}, & if\ x_i^t = Sbest_j^t \\ \dfrac{fitness\left(Sbest_{S_{(i)}^t}^t\right)}{fitness(x_i^t)}, & else \end{cases} \tag{4}$$

$N(t)$ represents the number of subswarms, $Sbest_j^t$ shows the best particle of the jth subswarm at the tth iteration, $S_{(i)}^t$ shows the subswarm with the position of the ith particle located at the tth iteration. Groups with different learning abilities also have different ways of updating their speed. Mathematically, the particle speed in the group with the first learning ability can be defined using the following model:

$$v_i^{t+1} = \omega_i^t v_i^t + c_{11} r_1 (Pbest_i^t - x_i^t) + c_{21} r_2 \left(AVGSbest^t - x_i^t\right), \tag{5}$$

where $c_{11} = 2$ and $c_{21} = 1$ denote positive acceleration coeeficients, r_1 and r_2 are two arbitrary numbers distributed uniformly on [0,1]. $Pbest_i^t$ denotes the best position of each particle and $AVGSbest^t$ is the average of the best positions of the particles in each subswarm which is defined as:

$$\begin{cases} AVGSbest_i^t = \sum_{j=1}^{N(t)-1} r_i^t * x_i^t \\ r_i^t = \dfrac{\left(1 - \dfrac{fitness(Sbest_i^t)}{\sum_{j=1}^{N(t)-1} fitness\left(Sbest_j^t\right)}\right)}{N(t)-2} \end{cases} \tag{6}$$

Furthermore, it was found:

$$\sum_{j=1}^{N(t)-1} r_i^t = 1 \tag{7}$$

The particle velocity in the group with the second learning ability is defined by the following equation:

$$v_i^{t+1} = \omega_i^t v_i^t + c_{12} r_1 (Pbest_i^t - x_i^t) + c_{22} r_2 \left(Sbest_{S(i)+1}^t - x_i^t\right), \tag{8}$$

where $c_{12} = 1$ and $c_{22} = 2$ are the acceleration coefficients and $Sbest_{S(i)+1}^t$ denote the learning characteristics of the best particle in the subswarm with a lower fitness level. Ordinary particles update their velocities using:

$$v_i^{t+1} = \omega_i^t v_i^t + c_{13} r_1 (Pbest_i^t - x_i^t) + c_{23} r_2 \left(Sbest_{S(i)+1}^t - x_i^t\right), \tag{9}$$

where $c_{13} = c_{23} = 1.5$ (Jiyue et al., 2023).

3. Human Learning Optimization (HLO)

The adaptive algorithm known as Human Learning Optimization (HLO) is based on activities of human learning. To master new things, humans repeatedly learn and practice them. After determining an appropriate learning method, humans can evaluate the recognition of a new object. The learning model used in HLO includes three learning stategy: random learning strategy, individual learning strategy, and social learning strategy (Wang, Ni, et al., 2015; Ding dan Gu 2020).

At the initial stage of a novel task, individuals typically lack prior knowledge, resulting in behavior characterized by trial and error. Within the framework of Human Learning Optimization (HLO), this phase is conceptualized as random learning, wherein actions are guided by uninformed guesses. As individuals accumulate experience through repeated attempts, they begin to discern effective strategies from ineffective ones based on personal outcomes. This process aligns with what is referred to in HLO as individual learning. Beyond self-experience, learning also occurs through observation and interaction with others. When individuals are exposed to peers with superior performance or greater experience, they are inclined to adopt or imitate those observed strategies. This behavior is captured in HLO as social learning, reflecting the influence of social context on cognitive adaptation (Wang, Ni, et al., 2015; Wang et al., 2018).

The three learning processes are determined based on the probability of random learning p_r and the probability of individual learning p_{in}. Once a random number $rand \in (0,1)$ is generated, the selection of the learning process is governed according to Equation (10) (Wang, Yang, et al., 2015).

$$\begin{cases} \text{individual learning,} & \text{if } rand \in (0, p_r) \\ \text{random learning,} & \text{if } rand \in [p_r, p_{in}] \\ \text{social learning,} & \text{else} \end{cases} \tag{10}$$

4. Modified Adaptive Particle Swarm Optimization-Human Learning Strategies (MAPSO-HLS)

The proposed algorithm is a combination of APSO-HSL and HLO called Modified Adaptive Particle Swarm Optimization-Human Learning Strategies (MAPSO-HLS). Every particle with a different role is given three types of learning. The learning process begins with individual learning, where each particle evaluates its own historical experience to refine movement strategies. This self-guided adaptation helps particles identify locally promising areas based on their own performance. Next, random learning introduces stochastic exploration, directs particle to study from K neighbors best experience, based on their previous experience. K neighbors used as learning references were randomly selected. Finally, social learning enables particles to adopt strategies from better-performing peers, promoting convergence toward globally optimal regions.

The speed and position particle-i at the $t+1$th iteration represented by v_i^{t+1} and x_i^{t+1} for the best particle in the entire swarm is given by Equation (11), for the best particle in each subswarm can be updated using Equation (12), and for ordinary particles can be updated using Equation (13). ω is the moment of inertia to balance local exploitation and global exploration calculated using Equation (4), r_1 and r_2 is any number uniformly distributed on $[0,1]$. c_{11}, c_{12}, c_{13} are the individual acceleration coefficients, and c_{21}, c_{22}, c_{23} are the social acceleration coefficients. $Pbest_i^t$ shows the best position of each particle and $AVGSbest^t$ is the average of the best positions of particles in each subswarm, g_K shows the best position from any K particle.

$$v_i^{t+1} = \begin{cases} \omega_i^t v_i^t + c_{11} r_1 (Pbest_i^t - x_i^t) + c_{21} r_2 \left(AVGSbest^t - x_i^t\right) p_{rn}; & p_{rn} \in (0, p_{in}) \\ \omega_i^t v_i^t + c_{11} r_1 (Pbest_i^t - x_i^t) + c_{21} r_2 (g_K - x_i^t); & p_{rn} \in [p_{in}, p_{sc}] \\ \omega_i^t v_i^t + c_{11} r_1 (Pbest_i^t - x_i^t)(1 - p_{sc}) + c_{21} r_2 \left(AVGSbest^t - x_i^t\right); & p_{rn} \in (p_{sc}, 1) \end{cases} \tag{11}$$

$$v_i^{t+1} = \begin{cases} \omega_i^t v_i^t + c_{12} r_1 (Pbest_i^t - x_i^t) + c_{22} r_2 \left(Sbest_{S(i)+1}^t - x_i^t\right) p_{rn}; & p_{rn} \in (0, p_{in}) \\ \omega_i^t v_i^t + c_{12} r_1 (Pbest_i^t - x_i^t) + c_{22} r_2 (g_K - x_i^t); & p_{rn} \in [p_{in}, p_{sc}] \\ \omega_i^t v_i^t + c_{12} r_1 (Pbest_i^t - x_i^t)(1 - p_{sc}) + c_{22} r_2 \left(Sbest_{S(i)+1}^t - x_i^t\right); & p_{rn} \in (p_{sc}, 1) \end{cases} \tag{12}$$

$$v_i^{t+1} = \begin{cases} \omega_i^t v_i^t + c_{13} r_1 (Pbest_i^t - x_i^t) + c_{23} r_2 \left(Sbest_{S(i)}^t - x_i^t\right) p_{rn}; & p_{rn} \in (0, p_{in}) \\ \omega_i^t v_i^t + c_{13} r_1 (Pbest_i^t - x_i^t) + c_{23} r_2 (g_K - x_i^t); & p_{rn} \in [p_{in}, p_{sc}] \\ \omega_i^t v_i^t + c_{13} r_1 (Pbest_i^t - x_i^t)(1 - p_{sc}) + c_{23} r_2 \left(Sbest_{S(i)}^t - x_i^t\right); & p_{rn} \in (p_{sc}, 1) \end{cases} \tag{13}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}, \tag{14}$$

Similar to HLO, the three learning strategies are independent of one another. Determination of the learning process based on the p_{in}, p_{sc}, and p_{rn}. The values $(p_{in} - 0)$ indicate the occurrence of individual learning processes, $(p_{sc} - p_{in})$ for random learning, and $(1 - p_{sc})$ for social learning, respectively. Parameter p_{rn} will be determined randomly at the interval (0,1).

5. Performance Evaluation

The performance of the modified algorithm was evaluated using multiple metrics, including the shortest distance, convergence rate, and diversity. A comparative analysis was conducted against the unmodified version of the algorithm using benchmark TSP instances. The comparison of the shortest distances was conducted by evaluating the exact solution against APSO-HSL and the exact solution against MAPSO-HLS using the percentage deviation PD shown by Equation (15).

$$PD = \frac{(\text{approximation} - \text{exact})}{\text{exact}} 100\% \tag{15}$$

The equation for determining diversity D shown by Equation (16).

$$D = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \text{Hamming}(x_i, x_j), \tag{16}$$

where N is the number of particles, x_i represents the position of the particle i in the dimensional space d. $\text{Hamming}(x_i, x_j)$ defined as the number of differing positions divided by the total length of the permutation. The factor $\frac{2}{N(N-1)}$ is used to calculate the average of all unique pair combinations without calculating them twice.

C. RESULT AND DISCUSSION

The chosen correlation coefficients were $c_{11} = 2$and $c_{21} = 1$ for the best particle in the entire swarm, $c_{12} = 1$ and $c_{22} = 2$ for the best particle in each subswarm, and $c_{13} = c_{23} = 1.5$ for ordinary particle. The learning behavior of each particle is determined by the probabilities $p_i = 0.4$ and $p_s = 0.8$. Each particle with free learning strategy learns from the best experience among $K = 5$ randomly selected neighbors. The number of particles used in this study was 100, with a total of 500 iterations. After being tested on six benchmark instances, MAPSO-HLS is compared with APSO-HSL based on the shortest route of exact solution. The comparison results of the shortest distances from six TSP datasets are presented in Table 1. A smaller percentage deviation indicates a higher accuracy of the algorithm in finding the optimal solution. As shown in Table 1, the percentage deviation of MAPSO-HLS are closer to the exact solutions compared to those of APSO-HSL.

Table 1. Comparison of APSO-HSL and MAPSO-HLS Results

Instances	Solution			Percentage Deviation (PD)	
	Exact	APSO-HSL	MAPSO-HLS	APSO-HSL	MAPSO-HLS
berlin52	7544.37	13236.05	12219.19	75.44%	61.96%
burma14	30.88	30.88	30.88	0.00%	0.00%
eil51	429.98	732.54	655.13	70.37%	52.36%
eil76	545.39	1151.24	1108.56	111.08%	103.26%
pr76	108159.44	207319.31	213169.12	91.68%	97.09%
st70	678.60	1612.76	1683.95	137.66%	148.15%
Average				81.04%	77.14%

The algorithm's performance was also evaluated based on its convergence behavior. In the context of the TSP, the convergence level of an algorithm is typically assessed through two main factors. The first is the speed at which the algorithm stabilizes or approaches its best-found solution over successive iterations. The second is the ability of the algorithm to converge toward the globally optimal solution, or in this case, the shortest possible route. These two aspects provide insight into both the efficiency and the accuracy of the optimization process. A visual comparison of the convergence patterns for MAPSO-HLS and APSO-HSL is presented in Figure 1 and Figure 2, respectively, highlighting differences in their convergence speed and final solution quality.

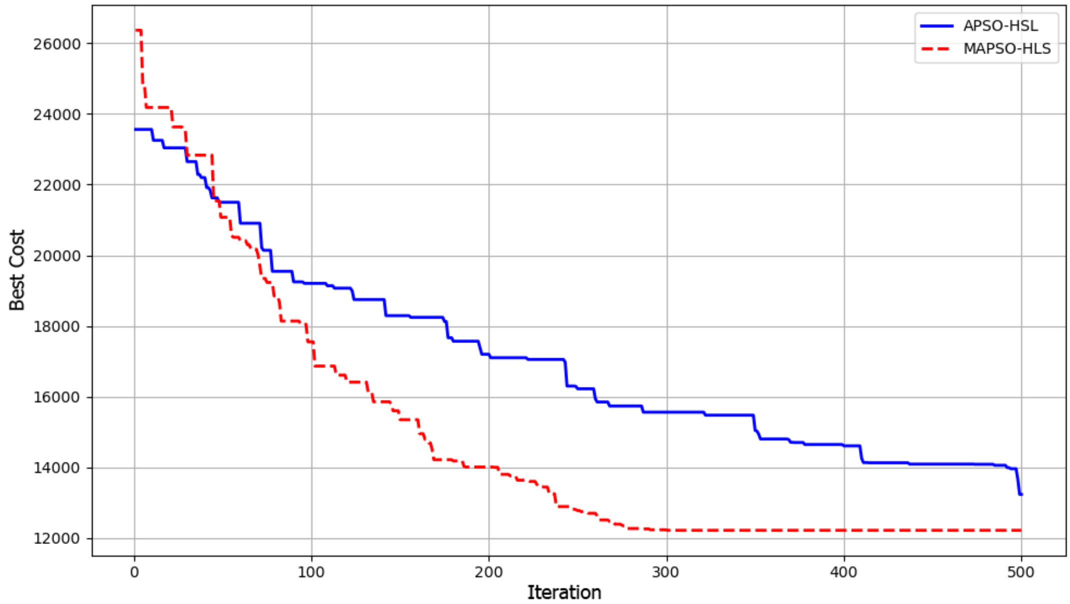

Figure 1. The Convergence Curve of berlin52

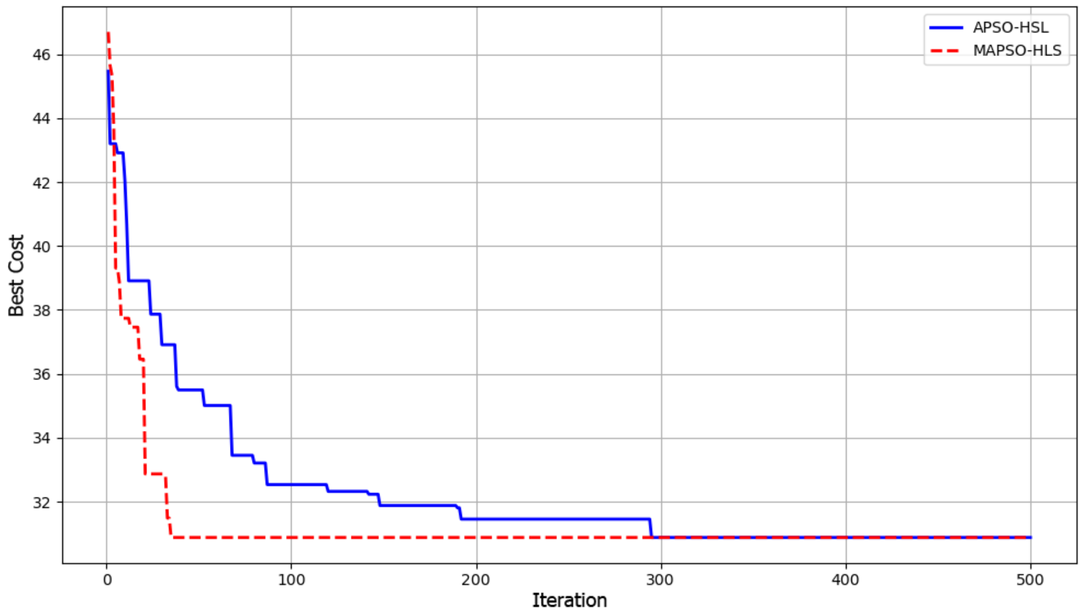

Figure 2. The Convergence Curve of burma14

As depicted in Figure 1, MAPSO-HLS begins to converge toward a lower objective value starting from iteration 300, indicating its effectiveness in refining solutions over time. Figure 2 further illustrates that MAPSO-HLS achieves convergence at an earlier stage, specifically before iteration 50, while APSO-HSL only starts to converge before iteration 300. This difference in convergence speed reflects the superior search efficiency of MAPSO-HLS. The earlier stabilization of MAPSO-HLS suggests that the algorithm is capable of rapidly identifying promising regions in the solution space and focusing its search efforts accordingly. In contrast, APSO-HSL requires more iterations to reach a similar level of solution quality, indicating a slower convergence process. Overall, these results confirm that MAPSO-HLS outperforms APSO-HSL in terms of convergence speed and solution refinement, making it a more effective approach for solving TSP problems in the tested scenarios.

In addition to the shortest route, and convergence, population diversity is also a key factor in determining the balance between exploration and exploitation in an algorithm. Exploration aims to avoid local optima and ensure that the entire solution space is thoroughly evaluated. Exploitation, on the other hand, focuses on refining promising solutions to become more optimal and accelerating convergence toward the optimal solution. The ideal condition, marked by a proper balance between exploration and exploitation, is characterized by high exploration in the early stages and high exploitation in the later stages. The diversity comparison is shown in Figure 3 and Figure 4.

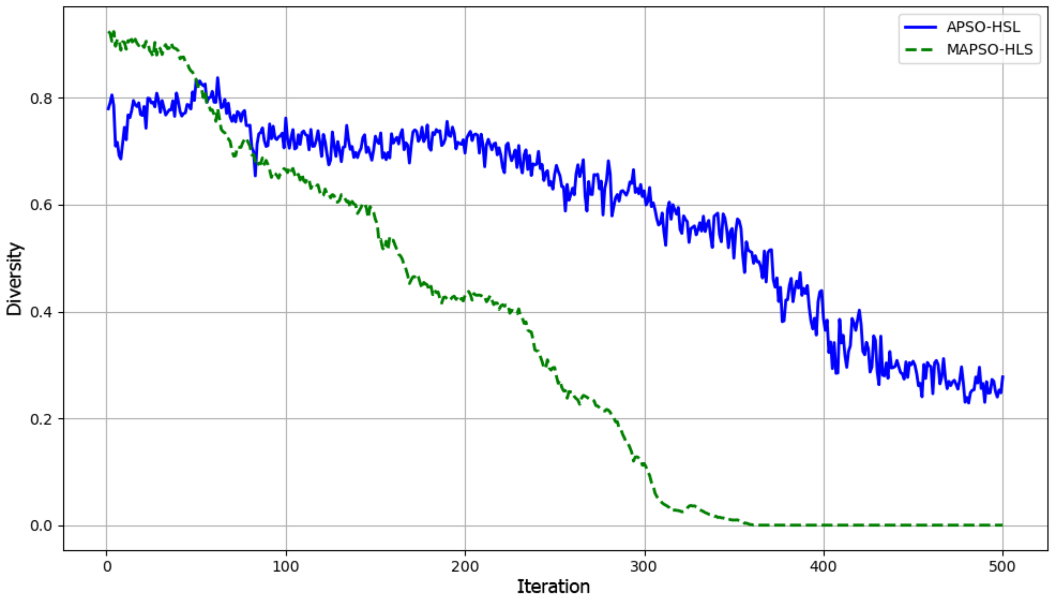

Figure 3 The Diversity Comparison of berlin52

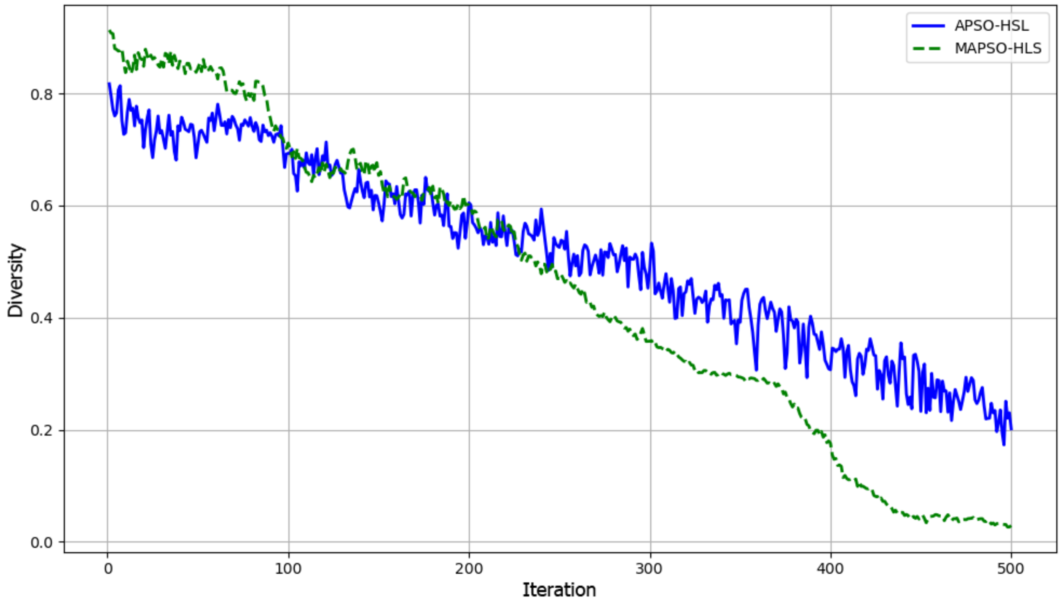

Figure 4 The Diversity Comparison of eil51

As illustrated in Figure 3, MAPSO-HLS demonstrates superior and more stable diversity throughout the optimization process. In the early iterations, the particles exhibit a high level of exploration by covering a wide range of solutions in the search space. This is reflected in the higher diversity values, indicating that the algorithm effectively avoids premature convergence. As the number of iterations increases, the particles gradually converge toward the optimal region, showing the algorithm's capability in intensifying the search or exploiting the promising areas. Figure 4 further confirms this behavior, where MAPSO-HLS maintains high diversity in the early phase and transitions smoothly into a more exploitative phase as the iteration progresses. Compared to APSO-HSL, MAPSO-HLS achieves a more balanced trade-off between exploration and exploitation, which is essential in avoiding local optima and enhancing solution quality. These findings highlight the effectiveness of MAPSO-HLS in maintaining population diversity while guiding the search toward optimal solutions.

The sequential application of three learning phases allows MAPSO-HLS to dynamically balance exploration and exploitation. Individual learning provides focused search at the particle level, random learning expands the search space to inject diversity, and social learning accelerates convergence by reinforcing successful behaviors. Compared to APSO-HSL, which relies solely on social learning, MAPSO-HLS demonstrates a more comprehensive and adaptive learning mechanism. This study contributes further by demonstrating the advantages of hybridizing APSO-HSL with human-inspired learning strategies in solving TSP problems.

D. CONCLUSION AND SUGGESTIONS

This study introduces MAPSO-HLS, a modified APSO-HSL algorithm that integrates human learning strategies, individual, random, and social learning, adapted from the HLO framework. The integration of these three learning mechanisms enhances the algorithm’s ability to balance exploration and exploitation. This balanced dynamic allows MAPSO-HLS to avoid premature convergence and more effectively navigate complex solution spaces. When applied to the Traveling Salesman Problem (TSP), MAPSO-HLS achieved an average deviation of 77.14%, while APSO-HSL recorded 81.04%, indicating that MAPSO-HLS is more effective in identifying shorter paths, with results closer to the exact solutions. MAPSO-HLS demonstrates superior convergence behavior, improved solution quality, and more stable population diversity, confirming its potential as a robust approach for solving combinatorial optimization problems.

Future research should incorporate statistical significance testing, extend comparisons to other PSO variants, and explore additional cognitive factors, such as motivation or memory decay, to further refine the learning mechanism. This approach offers a promising direction for developing intelligent optimization algorithms that more closely emulate human problem-solving behavior. Applying the algorithm to real-world datasets in logistics or scheduling could also validate its practical utility.

REFERENCES

Ab Wahab, M. N., Nefti-Meziani, S., & Atyabi, A. (2015). A comprehensive review of swarm optimization algorithms. *PLoS ONE*, *10*(5), 1-36. https://doi.org/10.1371/journal.pone.0122827

Al-Maamari, A., & Omara, F. A. (2015). Task scheduling using PSO algorithm in cloud computing environments. *International Journal of Grid and Distributed Computing*, *8*(5), 245–256. https://doi.org/10.14257/ijgdc.2015.8.5.24

Ashraf, A., Almazroi, A. A., Bangyal, W. H., & Alqarni, M. A. (2022). Particle swarm optimization with new initializing technique to solve global optimization problems. *Intelligent Automation and Soft Computing*, *31*(1), 191–206. https://doi.org/10.32604/IASC.2022.015810

Bangyal, W. H., Nisar, K., Soomro, T. R., Ag Ibrahim, A. A., Mallah, G. A., Hassan, N. U., & Rehman, N. U. (2023). An improved particle swarm optimization algorithm for data classification. *Applied Sciences (Switzerland)*, *13*(1), 1-18. https://doi.org/10.3390/app13010283

Chen, D., Imdahl, C., Lai, D., & Van Woensel, T. (2025). The Dynamic Traveling Salesman Problem with Time-Dependent and Stochastic travel times: A deep reinforcement learning approach. *Transportation Research Part C: Emerging Technologies*, *172*, 105022. https://doi.org/10.1016/J.TRC.2025.105022

Ding, H., & Gu, X. (2020). Hybrid of human learning optimization algorithm and particle swarm optimization algorithm with scheduling strategies for the flexible job-shop scheduling problem. *Neurocomputing*, *414*, 313–332. https://doi.org/10.1016/j.neucom.2020.07.004

Du, J., Wang, L., Fei, M., & Ilyas, M. (2022). A human learning optimization algorithm with competitive and cooperative learning. *Complex & Intelligent Systems*, *9*(1), 797-823. https://doi.org/ 10.1007/s40747-022-00808-4

Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a systematic review. *Archives of Computational Methods in Engineering*, *29*(5), 2531–2561. https://doi.org/10.1007/s11831-021-09694-4

Guo, W., Li, L., Chen, M., Ni, W., Wang, L., & Li, D. (2025). Balancing convergence and diversity preservation in dual search space for large scale particle swarm optimization. *Applied Soft Computing*, *169*, 112617. https://doi.org/10.1016/J.ASOC.2024.112617

Houssein, E. H., Gad, A. G., Hussain, K., & Suganthan, P. N. (2021). Major Advances in Particle Swarm Optimization: Theory, Analysis, and Application. *Swarm and Evolutionary Computation*, *63*, 100868. https://doi.org/10.1016/J.SWEVO.2021.100868

Jain, N. K., Nangia, U., & Jain, J. (2018). A review of particle swarm optimization. In *Journal of The Institution of Engineers (India): Series B* (Vol. 99, Issue 4, pp. 407–411). Springer. https://doi.org/10.1007/s40031-018-0323-y

Jarecki, J. B., Meder, B., & Nelson, J. D. (2018). Naïve and robust: class-conditional independence in human classification learning. *Cognitive Science*, *42*(1), 4–42. https://doi.org/10.1111/cogs.12496

Jedrzejowicz, P., Keller, K., Skakovski, A. (2024). An efficient hybrid evolutionary algorithm for solving the traveling salesman problem. *Procedia Computer Science*, *246(C)*, 3566-3574. https://doi.org/10.1016/j.procs.2024.09.201

Jiyue, E., Liu, J., & Wan, Z. (2023). A novel adaptive algorithm of particle swarm optimization based on the human social learning intelligence. *Swarm and Evolutionary Computation*, *80*, 101336. https://doi.org/10.1016/j.swevo.2023.101336

Larsen, R. B., Jouffroy, J., & Lassen, B. (2016). On the premature convergence of particle swarm optimization. *2016 European Control Conference, ECC 2016*, 1922–1927. https://doi.org/10.1109/ECC.2016.7810572

Lynn, N., & Suganthan, P. N. (2015). Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. *Swarm and Evolutionary Computation*, *24*, 11–24. https://doi.org/10.1016/J.SWEVO.2015.05.002

Piotrowski, A. P., Napiorkowski, J. J., & Piotrowska, A. E. (2020). Population size in Particle Swarm Optimization. *Swarm and Evolutionary Computation*, *58*(April), 100718. https://doi.org/10.1016/j.swevo.2020.100718

Punyakum, V., Sethanan, K., Nitisiri, K., Pitakaso, R., & Gen, M. (2022). Hybrid differential evolution and particle swarm optimization for multi-visit and multi-period workforce scheduling and routing problems. *Computers and Electronics in Agriculture*, *197*(2), 106929. https://doi.org/10.1016/j.compag.2022.106929

Ramdhani, L. S. (2016). Penerapan Particle Swarm Optimization (PSO) untuk seleksi atribut dalam meningkatkan akurasi prediksi diagnosis penyakit hepatitis dengan metode algoritma C4.5. *Swabumi*, *4*(1), 1–15. https://doi.org/10.31294/swabumi.v4i1.1011

Roberts-Mahoney, H., Means, A. J., & Garrison, M. J. (2016). Netflixing human capital development: personalized learning technology and the corporatization of K-12 education. *Journal of Education Policy*, *31*(4), 405–420. https://doi.org/10.1080/02680939.2015.1132774

Shaj, V., M, A. P., & S, A. (2016). Edge PSO: a recombination operator based PSO algorithm for solving TSP. In *Proceedings of the 2016 Inernational Conference on Advances in Computing, Communication and Informatics* (pp. 35–41). https://doi.org/10.1109/ICACCI.2016.7732022

TSPLIB. (n.d.). Retrieved November 3, 2024, from http://www.iwr.uniheidelberg.de/groups/comopt/software/TSPLIB95/tsp/

Vahdat, M., Oneto, L., Anguita, D., Funk, M., & Rauterberg, M. (2016). Can machine learning explain human learning? *Neurocomputing*, *192*, 14–28. https://doi.org/10.1016/j.neucom.2015.11.100

Wang, L., Ni, H., Yang, R., Pardalos, P. M., Du, X., & Fei, M. (2015). An adaptive simplified human learning optimization algorithm. *Information Sciences*, *320*(6), 126–139. https://doi.org/10.1016/J.INS.2015.05.022

Wang, L., Pei, J., Menhas, M. I., Pi, J., Fei, M., & Pardalos, P. M. (2017). A Hybrid-coded Human Learning Optimization for mixed-variable optimization problems. *Knowledge-Based Systems*, *127*(C), 114–125. https://doi.org/10.1016/j.knosys.2017.04.015

Wang, L., Pei, J., Wen, Y., Pi, J., Fei, M., & Pardalos, P. M. (2018). An improved adaptive human learning algorithm for engineering optimization. *Applied Soft Computing Journal*, *71*, 894–904. https://doi.org/10.1016/j.asoc.2018.07.051

Wang, L., Yang, R., Ni, H., Ye, W., Fei, M., & Pardalos, P. M. (2015). A human learning optimization algorithm and its application to multi-dimensional knapsack problems. *Applied Soft Computing Journal*, *34*, 736-743. https://doi.org/10.1016/j.asoc.2015.06.004

Zhong, Y., Lin, J., Wang, L., & Zhang, H. (2018). Discrete comprehensive learning particle swarm optimization algorithm with metropolis acceptance criterion for traveling salesman problem. *Swarm and Evolutionary Computation*, *42*(C), 77–88. https://doi.org/10.1016/j.swevo.2018.02.017