

Research on Intelligent Optimization Algorithm for Multi-Objective Vehicle Routing Based on Deep Reinforcement Learning

Xiangxin Wang

University of Shanghai for Science and Technology, Shanghai, 200093, China

Abstract: To address the problems of slow convergence and poor multi-objective coordination in complex scenarios of traditional multi-objective vehicle routing optimization algorithms, as well as the inaccurate state representation and policy update drift of existing deep reinforcement learning algorithms, this paper proposes an enhanced dynamic self-attention and adaptive reward-based PPO algorithm (ADAR-PPO) to solve the multi-objective vehicle routing optimization problem. First, a mathematical model for multi-objective vehicle routing optimization is constructed, clarifying optimization objectives such as shortest path, lowest transportation cost, and load balancing, along with constraints such as vehicle capacity and time windows. Second, an enhanced dynamic self-attention encoder is designed, integrating channel attention and SwiGLU activation units to achieve dynamic updating and deep fusion of node features, improving the accuracy of state representation. Then, an adaptive multi-objective reward mechanism is proposed, dynamically adjusting the weights of each objective and generating a comprehensive reward by combining constraint penalties to guide the agent in learning the Pareto optimal solution. Finally, the PPO policy update method is improved by introducing advantage normalization and dynamic pruning thresholds to enhance the algorithm's training stability and sample efficiency. Experimental results based on the Solomon standard dataset show that, in the C101 instance, the ADAR-PPO algorithm achieves an HV value of 0.896, a 12.8% improvement over the standard PPO and a 23.5% improvement over GA; an IGD value of 0.087, a 49.4% reduction over the standard PPO and a 61.3% reduction over PSO; an average path length of 892.3 m, a 12.6% reduction over GA; an average transportation cost of 702.5 yuan, a 7.6% reduction over PSO; and a convergence iteration count of 180, a 30% reduction over the standard PPO. These results demonstrate that the proposed algorithm outperforms the comparative algorithms in multi-objective collaborative optimization, convergence speed, solution accuracy, and perturbation adaptability, providing effective technical support for intelligent planning of logistics delivery routes.

Keywords: Deep Reinforcement Learning; Multi-objective Vehicle Routing Problem; ADAR-PPO Algorithm; Enhanced Self-attention; Adaptive Reward; PPO.

1. Introduction

The vehicle routing problem (VRP) is a core combinatorial optimization problem in logistics scheduling and urban delivery. Essentially, it aims to generate high-quality service paths for multiple vehicles under constraints such as capacity, time windows, and service order. With the fragmentation of e-commerce orders, the intensive delivery of same-city instant services, and the normalization of urban road congestion, traditional VRP modeling, which focuses solely on the shortest distance, is no longer sufficient to meet the needs of real-world systems [1]. The coupled optimization of objectives such as path length, transportation cost, vehicle load balancing, and customer timeliness has gradually become a research focus. Especially in complex delivery scenarios, the multi-objective VRP not only needs to consider multiple competing performance indicators but also requires algorithms to maintain high solution efficiency under dynamic constraints and high-dimensional state spaces, making it a key technical problem in intelligent logistics optimization [2]. Existing reviews and benchmark studies have shown that VRP and its time-window variants have long been important testing platforms for operations research optimization and intelligent decision-making. The 100-customer standard instance proposed by Solomon remains the mainstream benchmark for evaluating routing algorithm performance.

Traditional MVRP methods mainly include exact algorithms, genetic algorithms, particle swarm optimization, tabu search, and multi-objective evolutionary algorithms. These methods achieve good results in small- to medium-scale static instances, but their adaptability to high-dimensional constraints, multi-objective tradeoffs, and large-scale scenarios is limited [3]. On the one hand, heuristic rules are highly dependent on the problem structure and lack generalization ability; on the other hand, the multi-objective weighting methods are often fixed, making it difficult to dynamically coordinate the importance of different objectives according to the search stage. In recent years, deep reinforcement learning, through a closed-loop learning mechanism of "state-action-reward," has provided a unified framework for combinatorial optimization problems, from experience-driven to policy generation. PPO, due to its stable updates and simple implementation, is widely used for sequence decision-making; attention mechanisms, which can more effectively encode node relationships, have been used to solve various types of routing problems [4]. Meanwhile, learning-based methods for VRP have further developed from constructive decoding to improved search, hierarchical decision-making, and multi-agent collaboration.

Despite ongoing research, existing DRL solutions for MVRP still suffer from three main shortcomings. First, state encoding fails to adequately represent node demand, remaining capacity, time window slack, and local topological

relationships, leading to insensitivity of the policy network to key constraints [5]. Second, multi-objective rewards often employ static weighting and simple penalties, making it difficult to maintain reasonable search preferences in the early and later stages of training, and prone to the problem of one objective becoming overly dominant. Third, in combined decision-making scenarios, PPO may still experience policy update shifts due to fluctuations in the dominance scale and fixed pruning thresholds, affecting sample utilization and convergence stability. While recent VRP learning research has made progress in multi-vehicle, dynamic demand, random perturbation, and multi-objective scenarios, a unified design is still lacking in multi-objective collaboration, explicit constraint fusion, and stable policy updates.

Based on this, this paper proposes an ADAR-PPO algorithm for multi-objective vehicle path optimization. This paper first establishes a mathematical model of MVRP (Multi-Objective Vehicle Route Optimization) including path length, transportation cost, and load balancing; secondly, it designs an enhanced dynamic self-attention encoding module, improving state representation capability through channel attention and SwiGLU gating; thirdly, it constructs an adaptive multi-objective reward mechanism based on Tchebycheff scalarization to achieve dynamic adjustment of different objective weights; finally, it improves the PPO (Progressive Point of Purchase) update process by introducing advantage normalization, dynamic pruning threshold, and entropy regularization to enhance training stability [6]. The overall technical roadmap is: problem modeling—state encoding—reward design—policy

optimization—experimental verification. Through multiple comparative experiments on the Solomon standard dataset, the effectiveness of the proposed method in terms of convergence speed, solution set quality, and robustness is systematically verified.

2. Multi-Objective Vehicle Route Intelligent Optimization Algorithm Design

2.1. Overall Algorithm Architecture

ADAR-PPO consists of four core modules: environment interaction, state encoding, reward generation, and policy update. The environment interaction module generates a masked action set based on the current partial path, the remaining customer set, vehicle states, and time window feasibility. The state encoding module maps customer node features, vehicle state features, and the global path context into a high-dimensional representation [7]. The reward generation module outputs stage rewards based on the multi-objective changes and constraint violations of the current solution. The policy update module completes the joint iteration of the policy network and the value network based on the improved PPO. These four modules work collaboratively through a closed-loop mechanism of "feasible solution construction—performance feedback—policy correction," ensuring that policy learning is driven by both local decision quality and global Pareto direction constraints.

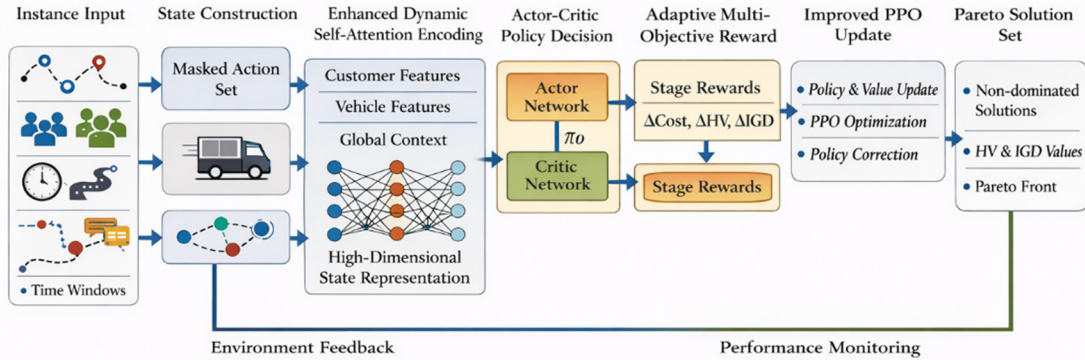


Figure 1. Overall Architecture of ADAR-PPO Algorithm

Let the delivery network be a directed graph $G = (V, E)$, where $V = \{0, 1, \dots, N\}$ represents the set of nodes, 0 is the delivery center, and E is the set of arcs; the vehicle set is $K = \{1, 2, \dots, M\}$. Customer i 's demand is q_i , service time is s_i , time window is $[e_i, l_i]$, and vehicle capacity is Q . If vehicle k travels from node i to node j , the decision variable $x_{ij}^k \in \{0, 1\}$ takes the value 1; otherwise, it takes the value 0. This paper formulates the problem as a three-objective joint optimization model:

$$\min_{\pi} F(\pi) = \{f_1(\pi), f_2(\pi), f_3(\pi)\}$$

$$\begin{cases} f_1(\pi) = \sum_{k=1}^M \sum_{i=0}^N \sum_{j=0}^N d_{ij} x_{ij}^k \\ f_2(\pi) = \sum_{k=1}^M \sum_{i=0}^N \sum_{j=0}^N (c_d d_{ij} + c_v) x_{ij}^k + c_w \sum_{i=1}^N \max(0, t_i - l_i) \\ f_3(\pi) = \sqrt{\frac{1}{M} \sum_{k=1}^M (L_k - \bar{L})^2} \end{cases} \quad (1)$$

In the formula, f_1 represents the total path length, and d_{ij} is the distance from node i to j ; f_2 represents the comprehensive transportation cost, where c_d is the unit distance cost, c_v is the fixed cost of vehicle dispatch, c_w is the time window penalty coefficient, and t_i is the service

time to customer i ; f_3 represents the vehicle load balancing deviation, L_k is the load rate of the k vehicle, and $\bar{L}$ is the average load rate of all vehicles. To ensure path feasibility, the following constraints must also be met: unique customer service, capacity constraint, and time window recursion constraint.

$$\begin{aligned} \sum_{k=1}^M \sum_{j=1}^N x_{ij}^k &= 1, \forall i \in V; \\ u_j^k &\geq u_i^k + q_j - Q(1 - x_{ij}^k), \\ t_j^k &\geq t_i^k + s_i + \tau_{ij} - \mathcal{M}(1 - x_{ij}^k), e_j \leq t_j^k \leq l_j. \end{aligned} \quad (2)$$

Where u_i^k represents the amount of cargo loaded when vehicle k leaves node i , τ_{ij} represents the travel time, and $\mathcal{M}$ is a sufficiently large relaxation constant. Equations (1) to (3) constitute the optimization objective and constraint basis in the ADAR-PPO environment.

2.2. Design of Enhanced Dynamic Self-Attention Encoding Module

To improve the state representation accuracy of multi-objective MVRP, this paper designs an enhanced dynamic self-attention encoder [8]. Unlike traditional attention mechanisms that only focus on static coordinate relationships, this module maps customer demand, remaining capacity, time window relaxation, and arc distance together into a dynamic association score. For the h attention head, the dynamic attention weight between node i and its neighboring node j at time t is defined as:

$$\alpha_{ij}^{h,t} = \frac{\exp\left(\frac{Q_i^{h,t}(K_j^{h,t})^\top}{\sqrt{d_h}} + \beta\Gamma_{ij}^t\right)}{\sum_{v \in \mathcal{N}_i} \exp\left(\frac{Q_i^{h,t}(K_v^{h,t})^\top}{\sqrt{d_h}} + \beta\Gamma_{iv}^t\right)} \quad (3)$$

Where $Q_i^{h,t}, K_j^{h,t}$ are the query vector and key vector, respectively, d_h is the single-head feature dimension, $\mathcal{N}_i$ is the candidate neighborhood, β is the dynamic relationship adjustment coefficient, and Γ_{ij}^t is the constraint-aware term used to fuse node demand ratio, time window relaxation, and distance normalization information. This term ensures that attention not only reflects semantic similarity but also the decision value of "whether it is worth visiting at this time."

In the aggregation stage, channel attention is introduced to perform secondary filtering on the multi-head output, resulting in an enhanced intermediate representation:

$$\tilde{h}_i^{t+1} = \text{LN}(h_i^t + [\oplus_{h=1}^H \sum_{j \in \mathcal{N}_i} \alpha_{ij}^{h,t} V_j^{h,t}] \odot c_i^t), c_i^t = \sigma(W_c \text{GAP}(H^t) + b_c) \quad (4)$$

In the formula, h_i^t is the input representation of node i at time t , $V_j^{h,t}$ is the value vector, $\oplus$ represents multi-head concatenation, $\odot$ represents channel multiplication, H^t is the hidden matrix of all nodes, $\text{GAP}(\cdot)$ represents global average pooling, W_c and b_c are channel mapping parameters, $\sigma(\cdot)$ is the Sigmoid function, and $\text{LN}(\cdot)$ is layer normalization. This structure can suppress invalid channel responses and enhance feature dimensions more relevant to the current path decision.

Subsequently, this paper introduces SwiGLU gated activation to form a deep nonlinear fusion:

$$h_i^{t+1} = W_o(\text{Swish}(W_1 \tilde{h}_i^{t+1}) \odot W_2 \tilde{h}_i^{t+1}) \quad (5)$$

Where W_1, W_2, W_o are learnable mapping matrices, and $\text{Swish}(x) = x \cdot \sigma(x)$. Equation (6) retains high-contribution features and weakens noise terms through gating, so that the encoding result retains both local constraint information and global context information. Based on Equations (4) and (6), the encoder can dynamically perceive the reachability and service priority between nodes, effectively alleviating the problem of insufficient description of complex multi-objective scenarios by traditional static coding.

2.3. Adaptive Multi-Objective Reward Generation Module Design

To avoid the rigid constraint of fixed weight rewards on the search direction at different training stages, this paper constructs an adaptive multi-objective reward function based on Tchebycheff scalarization. For the local solution at time t , the comprehensive reward is defined as:

$$r_t = - \max_{m \in \{1,2,3\}} \left[\lambda_m^{(t)} \left| \frac{f_m^{(t)} - z_m^*}{z_m^{\text{nad}} - z_m^* + \xi} \right| \right] - \eta p_t \quad (6)$$

Where $f_m^{(t)}$ is the observation value of the m target at

time t , z_m^* is the ideal point component, z_m^{nad} is the approximate Nadir point component, ξ is a minimal constant to prevent the denominator from being zero, $\lambda_m^{(t)}$ is the dynamic weight, η is the penalty term coefficient, and p_t is the constraint penalty. In this paper, $p_t = \omega_1 P_{\text{cap}} + \omega_2 P_{\text{tw}} + \omega_3 P_{\text{route}}$, where $P_{\text{cap}}, P_{\text{tw}}, P_{\text{route}}$ correspond to capacity violation, time window violation, and illegal loop penalty, respectively, and $\omega_1, \omega_2, \omega_3$ are their weights.

To ensure that the reward automatically focuses on targets with smaller improvement or more significant degradation at different stages of the search, the dynamic weight is designed as follows:

$$\lambda_m^{(t)} = \frac{\exp(\gamma \delta_m^{(t)})}{\sum_{u=1}^3 \exp(\gamma \delta_u^{(t)})}, \delta_m^{(t)} = \frac{\bar{f}_m^{(t)} - \bar{f}_m^{(t-1)}}{|\bar{f}_m^{(t-1)}| + \xi} \quad (7)$$

Where $\bar{f}_m^{(t)}$ represents the moving average of the m objective in the current batch, $\delta_m^{(t)}$ represents the rate of change of the objective, and γ is the temperature coefficient. If an objective has not improved sufficiently recently, its $\delta_m^{(t)}$ will increase, and the corresponding weight $\lambda_m^{(t)}$ will rise, thereby driving the policy to refocus on that objective. This mechanism transforms the reward function from a "fixed preference" to a "stage-adjusted preference," which is more conducive to learning Pareto equilibrium solutions.

2.4. Improved PPO Policy Update Module Design

In the PPO update stage, this paper first normalizes the advantage function to reduce gradient fluctuations caused by differences in reward scales between different batches:

$$\hat{A}_t = \frac{A_t - \mu_A}{\sigma_A + \varepsilon}, \epsilon_t = \text{clip}(\epsilon_{\max} e^{-\kappa \bar{D}_t}, \epsilon_{\min}, \epsilon_{\max}) \quad (8)$$

In the formula, A_t is the generalized dominance estimate, μ_A and σ_A are the mean and standard deviation of the current batch dominance, respectively, ε is the numerical stability constant; ϵ_t is the dynamic pruning threshold, $\epsilon_{\max}$ and $\epsilon_{\min}$ are the upper and lower bounds, respectively, κ is the decay coefficient, and $\bar{D}_t$ is the average KL offset between the old and new policies in the current batch [9]. This design automatically shrinks the pruning interval when the policy changes too rapidly; when training is relatively stable, the policy update step size is appropriately widened, thus balancing stability and search efficiency.

The final policy optimization objective is defined as:

$$L(\theta) = \mathbb{E}_t[\min(\rho_t(\theta) \hat{A}_t, \text{clip}(\rho_t(\theta), 1 - \epsilon_t, 1 + \epsilon_t) \hat{A}_t) + c_e \mathcal{H}(\pi_\theta(\cdot | s_t))] - c_v \mathbb{E}_t[(V_\theta(s_t) - R(t^2 \mathbb{0}))] \quad (9)$$

Where $\rho_t(\theta) = \pi_\theta(a_t | s_t) / \pi_{\theta_{\text{old}}}(a_t | s_t)$ is the probability ratio, $\mathcal{H}(\pi_\theta)$ is the policy entropy, c_e is the entropy reward coefficient, c_v is the value loss coefficient, $V_\theta(s_t)$ is the value network output, and R_t is the cumulative reward. Equation (10) adds advantage normalization and dynamic pruning mechanism to the classic PPO pruning objective, which can effectively suppress policy drift; the entropy term improves the diversity of early search and enhances the ability to escape local optima.

3. Experimental Simulation and Result Analysis

3.1. Experimental Environment Setup

The experimental platform uses Ubuntu 22.04 operating system, with an AMD Ryzen 9 7950X CPU and an NVIDIA

RTX 4090 GPU with 24 GB of video memory; the software environment is Python 3.10, PyTorch 2.2, and CUDA 12.1. To verify the algorithm's adaptability under different spatial distributions, three typical instances from the Solomon standard dataset—C101, R101, and RC101—were selected as the core test objects [10]. This benchmark originates from Solomon's 1987 VRPTW study, containing 56 instances of 100 customers each. Class C represents a clustered customer distribution, Class R a random distribution, and Class RC a mixed distribution, effectively covering typical scenarios in urban delivery.

In the environmental state construction, customer node features include two-dimensional coordinates, demand, service time, upper and lower bounds of the time window, and whether the service has been marked; vehicle state features include current load, current location, distance traveled, remaining available time, and number of customers currently served [11]. To evaluate the quality of the multi-objective solution set, this paper uses the hypervolume index HV and the inverse generation distance IGD as Pareto performance evaluation metrics, and additionally calculates the average path length, average transportation cost, and number of convergence iterations. Considering the differences in the dimensions of different metrics, the three objective values are first normalized to minimum and maximum values in the experiment, and then the reference point is set as (1.20, 1.20, 1.20) (1.20, 1.20, 1.20) (1.20, 1.20, 1.20).

3.2. Comparison of Algorithms and Parameter Settings

To ensure sufficient comparison, four representative algorithms were selected as baselines: standard PPO, genetic algorithm (GA), particle swarm optimization (PSO), and NSGA-II. Standard PPO uses the same Actor-Critic skeleton as presented in this paper, but does not introduce enhanced dynamic self-attention, adaptive reward, or dynamic pruning mechanisms; GA uses sequential encoding and crossover-mutation search; PSO uses discrete particle position mapping; and NSGA-II, as a typical multi-objective evolutionary method, was used to compare Pareto front quality.

Regarding parameters, ADAR-PPO was configured with 3 encoding layers, 8 attention heads, 128 hidden dimensions, a learning rate of 3×10^{-4} , a discount factor of 0.99, a GAE parameter of 0.95, a batch size of 64, an entropy coefficient of 0.01, and $\epsilon_{\min}[f_0]=0.08$ and $\epsilon_{\max}[f_0]=0.22$. GA, PSO, and NSGA-II were all configured with a population size of 100 and a maximum number of iterations of 300 to ensure a

consistent computational budget. All algorithms were run independently 20 times on the same hardware platform, and the average result was used as the final statistical value.

3.3. Experimental Indicators and Testing Scheme

This paper sets five core evaluation metrics: HV (Hypervised Value), IGD (Increased Income Difference), average path length, average transportation cost, and convergence iterations. A higher HV indicates better Pareto solution set coverage, while a lower IGD indicates the solution set is closer to the true frontier. Average path length and average transportation cost reflect scheduling efficiency and economy, respectively, and the number of convergence iterations reflects training speed. In addition, load standard deviation is recorded as a supplementary metric to aid in analyzing vehicle utilization balance.

The test scheme is divided into three groups. Scheme A is the standard scenario test, comparing the basic performance of each algorithm on C101, R101, and RC101 instances. Scheme B is the demand fluctuation scenario test, adding random disturbances of 5%, 10%, and 15% to customer demand on top of the original values. Scheme C is the fault disturbance test, introducing vehicle fault probabilities of 0.03, 0.06, and 0.09 during the path execution phase to trigger local replanning. These three test groups correspond to the three dimensions of "static performance—data disturbance adaptability—operational anomaly robustness," providing a comprehensive evaluation of the algorithm's engineering usability.

3.4. Experimental Results Analysis

3.4.1. Comprehensive Comparison of Standard Scenario

Table 1 presents the comprehensive results of different algorithms in the C101 instance. It can be seen that ADAR-PPO achieves the best results in HV, IGD, average path length, average transportation cost, and number of convergence iterations. Especially in terms of solution set quality indicators, HV reaches 0.896, a significant improvement over the standard PPO; IGD decreases to 0.087, indicating that the obtained Pareto solution set distribution is more uniform and has a higher approximation degree. The simultaneous decrease in path length and transportation cost indicates that this method does not sacrifice one objective for the improvement of another, but rather achieves better multi-objective collaborative optimization.

Table 1. Comprehensive Performance Comparison of Different Algorithms in the C101 Instance

Algorithm	HV	IGD	Average path length / m	Average transportation cost / yuan	Load standard deviation	Convergence iterations / generation
GA	0.726	0.201	1021.0	763.4	0.236	318
PSO	0.754	0.225	955.8	760.3	0.214	286
NSGA-II	0.821	0.119	918.7	721.4	0.162	231
Standard PPO	0.794	0.172	940.6	745.2	0.187	257
ADAR-PPO	0.896	0.087	892.3	702.5	0.121	180

Table 1 further shows that ADAR-PPO shortens the average path length by 128.7 m compared to GA, reduces transportation costs by 57.8 yuan compared to PSO, and has a significantly smaller load standard deviation, indicating that its generated vehicle task allocation is more balanced. This result shows that the enhanced coding module does indeed improve the ability to identify node and constraint

relationships, while the adaptive reward effectively strengthens the collaborative advancement among multiple objectives.

3.4.2. Convergence Characteristics Analysis

Figure 2 shows the HV convergence curves of each algorithm on the C101 instance. ADAR-PPO has formed a significant lead advantage after about 50 generations and

tends to stabilize around the 180th generation; in contrast, standard PPO still shows significant fluctuations after 200 generations, while GA and PSO have a slower overall growth rate. This phenomenon indicates that the advantage normalization and dynamic pruning threshold introduced in this paper can more effectively control the policy update amplitude and avoid training oscillations caused by policy mutations. From the curve slope, ADAR-PPO has stronger exploration efficiency in the early stage and can maintain

good convergence smoothness in the middle and late stages. This means that the proposed algorithm can quickly identify high-value customer access patterns in the early stages of training, and then suppress ineffective large-step updates through dynamic pruning in the later stages, thereby achieving the optimization characteristics of "fast start-up and stable convergence". For real-world logistics scenarios, this characteristic directly corresponds to lower training overhead and faster strategy deployment speed.

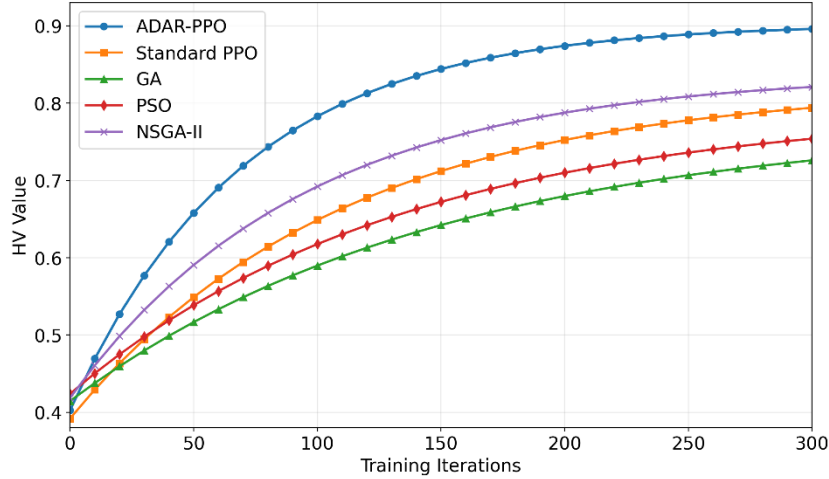


Figure 2. HV Convergence Curves of Algorithms in C101 Example

3.4.3. Pareto Front Distribution Analysis

Figure 3 shows the solution set distribution of different algorithms in the dual-objective plane of "average path length - average transportation cost". It can be seen that the solution points generated by ADAR-PPO are generally concentrated in the lower left region, indicating that it can simultaneously obtain shorter paths and lower costs; at the same time, the solution points are relatively evenly distributed along the front direction, indicating that it has good coverage of solutions with different preferences. In contrast, the solution sets of GA and PSO are more dispersed, and there are many dominated solutions; although standard PPO has some competitiveness in local regions, the front boundary is not

clear enough, indicating that its strategy has a bias among multiple objectives.

There are two main reasons for this phenomenon: First, the enhanced dynamic self-attention encoder explicitly injects information on time window relaxation, demand ratio, and distance constraints, so that the strategy no longer relies solely on local distance when selecting the next customer; second, the adaptive reward based on Tchebycheff dynamically increases the optimization weight of the "weakest link objective" during training, thereby suppressing the long-term dominance of a single objective on the search direction. The results show that ADAR-PPO not only approaches the Pareto front, but also improves the balance of the solution distribution.

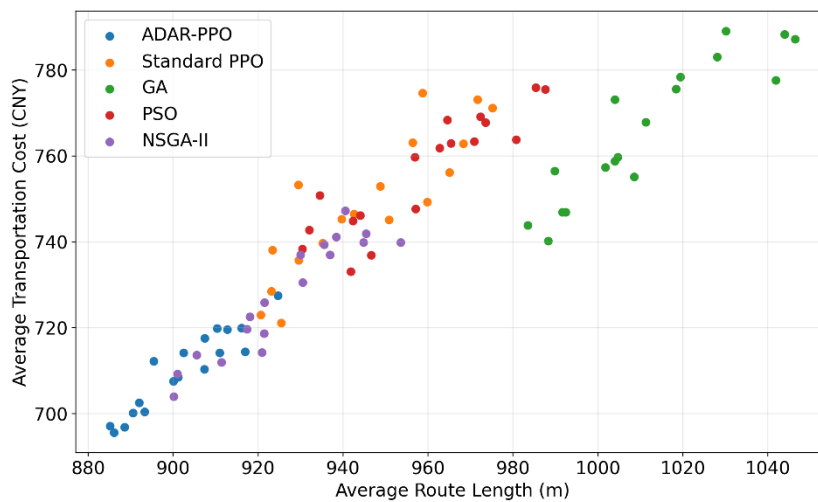


Figure 3. Pareto distribution of different algorithms on the distance-cost plane

4. Conclusion

This paper proposes an ADAR-PPO algorithm based on enhanced dynamic self-attention and adaptive reward mechanism for the multi-objective vehicle routing optimization problem. By constructing a multi-objective model including path length, transportation cost, and load balancing, a state encoder integrating channel attention and SwiGLU is designed, a dynamic reward function based on Tchebycheff scalarization is established, and a PPO update strategy of advantage normalization and dynamic pruning threshold is introduced, forming a unified intelligent optimization framework for complex logistics route planning. Experimental results show that ADAR-PPO achieves superior overall performance compared to GA, PSO, NSGA-II, and standard PPO on the Solomon standard dataset, demonstrating outstanding performance in HV, IGD, average path length, transportation cost, and convergence speed. It also exhibits good robustness under demand fluctuations and vehicle fault disturbances.

References

- [1] Li, K. W., Zhang, T., Wang, R., Qin, W. J., He, H. H., & Huang, H. (2021). Research progress in combinatorial optimization based on deep reinforcement learning. *Acta Automatica Sinica*, 47(11), 2521–2537.
- [2] Zhang, D. Z., Qiao, X., Xiao, B. W., Wang, R. D., & Mao, C. H. (2021). Multi-objective vehicle routing optimization based on low-carbon and stochastic demand. *Journal of Railway Science and Engineering*, 18(8), 2165–2174.
- [3] Wang, W. L., Chen, H. L., Li, G. Q., Leng, L. L., & Zhao, Y. W. (2022). Vehicle routing planning for multiple distribution centers based on deep reinforcement learning. *Control and Decision*, 37(8), 2101–2109.
- [4] Liu, Y. Q., & Han, J. (2022). Model and algorithm for vehicle routing problem of freight cars carrying multiple UAVs. *Journal of Transportation Engineering and Information*, 20(3), 102–113.
- [5] Hou, D. K., Fan, H. M., & Ren, X. X. (2022). Optimization of vehicle routes for joint delivery by multi-center hybrid fleets under time-varying road networks. *Journal of Dalian Maritime University*, 48(1), 11–22.
- [6] Li, Y., Fan, H. M., & Zhang, X. N. (2022). Periodic optimization model and solution for vehicle routing problem under dynamic demand. *Chinese Journal of Management Science*, 30(8), 254–266.
- [7] Fan, H. M., Yang, C., Zhang, Y. G., Sun, X. N., & Tian, P. J. (2023). Research on vehicle route optimization of multi-center hybrid fleets under mixed time windows. *Computer Integrated Manufacturing Systems*, 29(10), 3529–3546.
- [8] Li, J. Q., Cai, J. C., Sun, T., Zhu, Q. L., & Lin, Q. Z. (2024). Multi-task assisted evolutionary algorithm for vehicle routing planning in complex logistics and distribution scenarios. *Acta Automatica Sinica*, 50(3), 544–559.
- [9] Zhang, X. Y., Jin, P., Hu, X. X., & Zhu, M. N. (2024). Research on open vehicle routing problem with time window for time-dependent multi-distribution center. *Chinese Journal of Management Science*, 32(1), 146–157.
- [10] Jiang, M., & He, T. (2025). Solving vehicle routing problem with capacity constraints based on deep reinforcement learning. *Journal of System Simulation*, 37(9), 2177–2187.
- [11] Zhang, Y. X., Lei, B. B., Wang, X. F., Zhu, X. J., & Song, J. H. (2025). A review of meta-heuristic algorithms for vehicle routing problem with time window. *Journal of Computer Applications Research*, 42(5), 1290–1298.