Adaptive Distributed k-WTA Network Under Time-varying Topologies for Multi-AGV Automated Loading Systems

Qiuyu Xu[1], Zhengtao Zhang[2], Fei Chen[3], Haiming Dong[4], Zuoxun Wang[1,*], Jianjun Jiang[4]
[1]Shandong Technology and Business University, Yantai 264005, China
[2]Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[3]Shandong Electrical Engineering & Equipment Group Co., Ltd., Jinan 250022, China
[4]Shandong Haiguan Electric Co., Ltd., Jinan 250101, China
*Corresponding email: 202214078@sdtbu.edu.cn

Abstract

To address the multi-task allocation problem of collaborative multi-automated guided vehicle (AGV) operations in automated loading systems, an adaptive multi-feature distributed k-Winners-Take-All (k-WTA) task allocation method is proposed. By integrating information such as AGV position, payload, battery level, and operational capacity, a task fitness evaluation model is constructed. Furthermore, by incorporating time-varying communication topologies and a data-driven adaptive gain, rapid and stable convergence is achieved in dynamic environments. A Power-arctan activation function is specifically designed to enhance the noise immunity of the system. Simulation results demonstrate that the proposed method can effectively suppress dynamic state oscillations, maintain fast and stable convergence performance under various topologies and swarm sizes, and realize effective multi-AGV task allocation.

Keywords

Automated guided vehicle, Task allocation, Time-varying communication topology, Distributed k-Winners-Take-All

Introduction

Systems such as multi-AGVs are deployed in various applications, including warehouse management, industrial factories, aviation logistics, and the food and beverage (F&B) industry. Compared to single-AGV operations, multi-AGV systems can enhance overall system efficiency through the collaborative execution of multiple transportation tasks [1]. However, as the number of AGVs and the scale of tasks increase, rational and efficient task allocation becomes a critical factor affecting system performance.

To address this challenge, numerous solution methods have been put forward in existing research. These approaches cover a wide range of categories: market and auction mechanisms, game theory approaches, heuristic and swarm intelligence algorithms, machine learning and self-organizing maps (SOM), coalition formation and collaborative planning, and neural networks together with k-WTA competition. In terms of market mechanisms and auction algorithms, researchers address the mapping between resources and tasks by simulating the bidding process.

For example, in edge computing and multi-agent task offloading, Wang et al. proposed a truthful auction resource allocation mechanism with a flexible task offloading scheme [2]. Zhang Lianming et al. developed a joint scheduling algorithm based on double auctions and mobility awareness [3]. For multi-robot systems, Choi et al. designed the Consensus-based Bundle Algorithm (CBBA) to achieve decentralized and robust allocation [4]. Furthermore, related studies have developed allocation algorithms based on anxiety models and improved contract net protocols, centralized allocation modeling for unmanned aerial vehicle (UAV) swarms, agent-based dynamic Nectar allocation algorithms, and task allocation methods based on combinatorial bidding [5-8]. These methods are logically intuitive, but in large-scale AGV automated loading scenarios can easily lead to a surge in communication overhead and decision delays.

To achieve adaptive task allocation in dynamic environments, some scholars have turned to game theory and reinforcement learning, driving the system to achieve a Nash equilibrium by constructing payoff models among agents. Zhang Meiyan et al. utilized game-theoretic approaches to solve the target hunting task allocation problem for heterogeneous formations [9]. Huo et al. proposed a game-theoretic task allocation framework for multi-mobile robot systems with minimum requirements [10]. To further improve solving efficiency, Xing et al. combined reinforcement learning with potential game models to achieve efficient multi-agent task allocation [11]. Similar mechanisms also include anonymous hedonic game models for large-scale systems and potential game frameworks for handling dynamic tasks [12,13]. Although game theory possesses excellent self-organizing properties, in highly dynamic warehousing environments, solving for equilibrium states often requires a long iteration time, making it difficult to meet the demands of instantaneous scheduling.

Facing practical loading tasks involving complex constraints such as load capacity, distance, and deadlines, heuristic algorithms, machine learning, and coalition formation strategies have been widely applied. At the machine learning algorithm level, J. Wu et al. combined an improved self-organizing map (ISOM) with an attention mechanism for dynamic allocation, while Y. Liu and S. Ma utilized SOM combined with the fast-marching method and unsupervised k-means learning strategies, respectively to optimize task execution order [14-16]. At the heuristic algorithm level, research covers multi-task allocation using genetic algorithms, an ant colony-parthenogenetic genetic hybrid algorithm (AC-PGA) for handling large-scale traveling salesman problems, capacity-constrained pickup and delivery task allocation integrated with path planning, and distributed deadline-constrained algorithms [17-20].

Furthermore, researchers have proposed a variety of coalition formation and hierarchical strategies to execute tightly coupled tasks. These strategies include communication-aware distributed coalitions, hierarchical strategies that guarantee distance optimality, reallocation methods based on heuristic cost evaluation, and heterogeneous robot coalitions built with the IQ-ASyMTRe framework and combinatorial bidding mechanisms [21-26]. Although such methods possess strong global search capabilities, their computational complexity is usually high, which heavily consumes the computing resources of the underlying hardware.

To completely overcome the computational bottlenecks of traditional algorithms in high-frequency dynamic scheduling, competitive neural networks based on k-WTA (k-Winners-Take-All) have stood out due to their extremely low computational complexity and distributed solving characteristics. Liu and Wang pioneered the conversion of k-WTA operations into quadratic programming problems and solved them using dual neural networks, laying the theoretical foundation for this field [27].

Subsequently, Qi et al. generated robust k-WTA networks and applied them to multi-agent coordination, while S. Liu et al. proposed a gradient-based differential k-WTA network to solve multi-robot competitive cooperation problems [28,29]. In the evolution of distributed architecture, Y. Zhang et al. designed a distributed k-WTA network from an optimization perspective [30]. K. Liu et al. evaluated the task allocation performance of this network on dynamic undirected connected graphs [31]. M. Liu et al. further constructed a TDCI-kWTA network considering communication interference and time delays [32].

These research breakthroughs demonstrate that continuous-time dynamic networks have prominent advantages in speed and decentralization for distributed decision-making. This finding also lays a solid theoretical foundation for our work. Specifically, this paper design a noise-tolerant multi-feature N-k-WTA automated loading task allocation framework for AGV swarms. The aforementioned k-WTA networks have shown great potential in computational efficiency and decentralized architecture.

However, they face tough challenges when applied to AGV swarms for automated loading. These challenges include highly heterogeneous physical constraints and high sensitivity of electromechanical systems. The physical constraints cover load status, available battery level, and dynamic A* planning distance. The underlying electromechanical systems are extremely sensitive to high-frequency environmental noise.

For these reasons, existing models commonly have several defects: limited input dimensions, poor topology adaptability, and insufficient noise tolerance. To bridge this research gap, this paper proposes an automated loading AGV task allocation framework integrating distributed state heterogeneous scalarization (DSHS) and noise-tolerant dynamic N-k-WTA. The specific design consists of the following three core modules.

(1) A multi-feature adaptation layer for the decision frontend is designed. By introducing a task pattern matrix and a dynamic weight adjustment mechanism, this layer uniformly maps the high-dimensional heterogeneous states of AGVs with inconsistent physical dimensions into a low-dimensional decision priority scalar.
(2) A data-driven adaptive gain control strategy based on time-varying topologies is designed. It switches topology states according to time and dynamically adjusts the convergence step size based on system state errors, achieving a coordination between fast response and stable convergence.
(3) A novel Power-Arctan activation function is designed to mitigate the impact of system input noise on the system.
The overall system design diagram is shown in Figure 1.

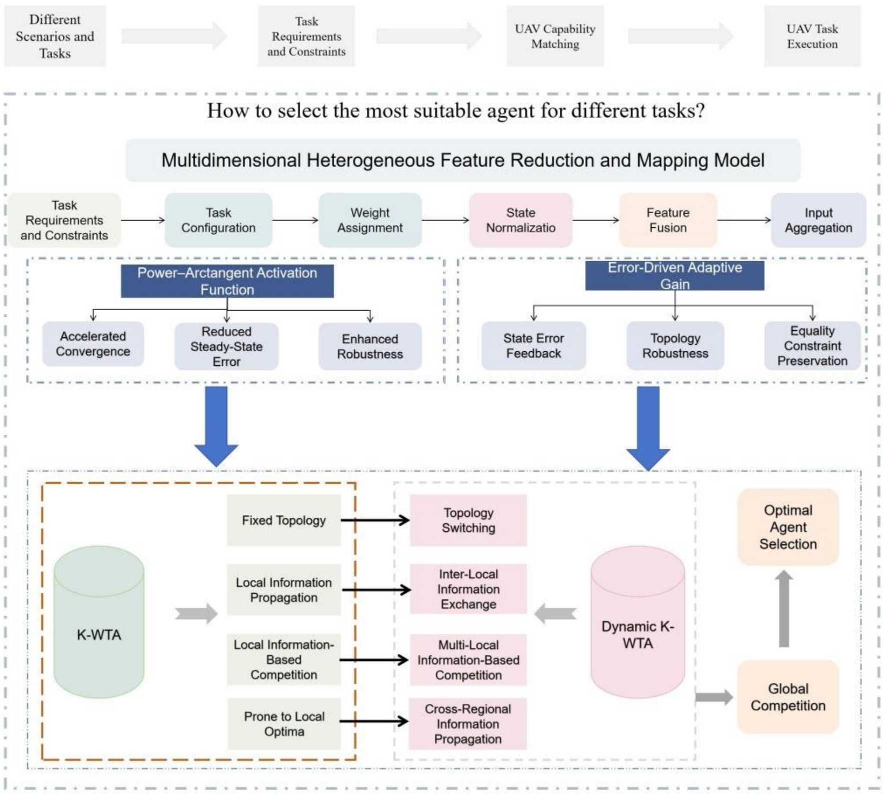


Figure 1. System architecture diagram.

Multi-state heterogeneous scalar layer

For a multi-robot system comprising m AGVs, the position of the i-th AGV is defined as $v_i \in R^2$, where $i \in \{1,2,3,\cdots,m\}$. In this paper, v_i is a two-dimensional vector, whose two elements represent the X-coordinate and Y-coordinate of the i-th AGV, respectively. In addition, the position of the target AGV is defined as $v_0 \in R^2$.
Therefore, this paper can define a variable to represent the distance between the i-th AGV and the target, whose expression is: $r_i(t) = -||v_i(t) - v_0(t)||$, which can take the distance $r_i(t)$ as the input, and model it as the k-WTA problem based on this input.
In actual collaborative task scenarios, the task decision-making of AGVs is influenced not only by spatial distance but also closely related to multi-dimensional heterogeneous states such as remaining battery level, environmental risk, and task load.
Due to differences in dimensions and physical meanings among various state variables, it is difficult to directly use them for task competition in multi-AGV collaborative scenarios. These state variables cover motion status, remaining energy, and load capacity of each vehicle, which carry distinct units and evaluation criteria. Therefore, unified modeling and fusion are required to construct a scalar evaluation index that can comprehensively reflect the task adaptability of AGVs under complex working conditions.
On this basis, a time-varying communication topology is introduced to construct a distributed k-WTA competitive network. This network allows intermittent information exchange among agents, enabling AGVs to complete

task selection and allocation in dynamic communication environments with unstable signal connections and changing neighbor sets.

TACMC-adaptation layer model

In the distributed k-WTA network, the input variable r_i of each agent is used to represent its priority or task adaptability in the current task. The larger the input value, the higher the probability that the agent will be activated and become a winner during the competition process. This paper designs a distributed state heterogeneous scalarization model (TACMC-adaptation layer). The core idea of this model is to construct a nonlinear mapping mechanism from the high-dimensional heterogeneous physical sensing space to the low-dimensional scalar decision priority manifold, which is used to adaptively generate the input variable of the k-WTA network according to task types and constraint conditions.

First, define the task mode selection matrix A_i. Acting as a dimension aligner, A_i dynamically screens active constraints and eliminates redundant perturbation terms according to the current task mode. Its row-column structure is determined by the task type i, and it is used to filter the state indicators participating in the calculation. Its expression is:

$$A_i = \begin{matrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1_{c\times c} \end{matrix} \tag{1}$$

The system introduces a dynamic weight modulation matrix to reflect the sensitivity of the agent to specific constraints. According to different task requirements, weights are assigned to each state to construct a weight matrix, which is a diagonal matrix whose diagonal elements represent the importance of each state under the current task. Its expression is:

$$\omega_i = \begin{matrix} \omega_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \omega_{i_{c\times c}} \end{matrix} \tag{2}$$

Further, the task-driven mapping operator z_i is obtained:

$$z_i = A_i \omega_i \tag{3}$$

Subsequently, the local utility state vector T_i is introduced into the above result:

$$T_i = \begin{bmatrix} T_{i,1} \\ T_{i,2} \\ T_{i,3} \\ \vdots \\ T_{i,c} \end{bmatrix}_{c\times 1} \tag{4}$$

The decision manifold projection component is obtained, with the expression as follows:

$$w_i = A_i \omega_i T_i \tag{5}$$

Finally, by summing the contributions of each state, the comprehensive priority scalar r_i of the agent is obtained:

$$r_i = 1^T \sum_{i=1}^{i} w_i \tag{6}$$

The overall structure is shown in Figure 2. Based on the selected, the model computes its comprehensive value for all idle agents to select the winning agents.

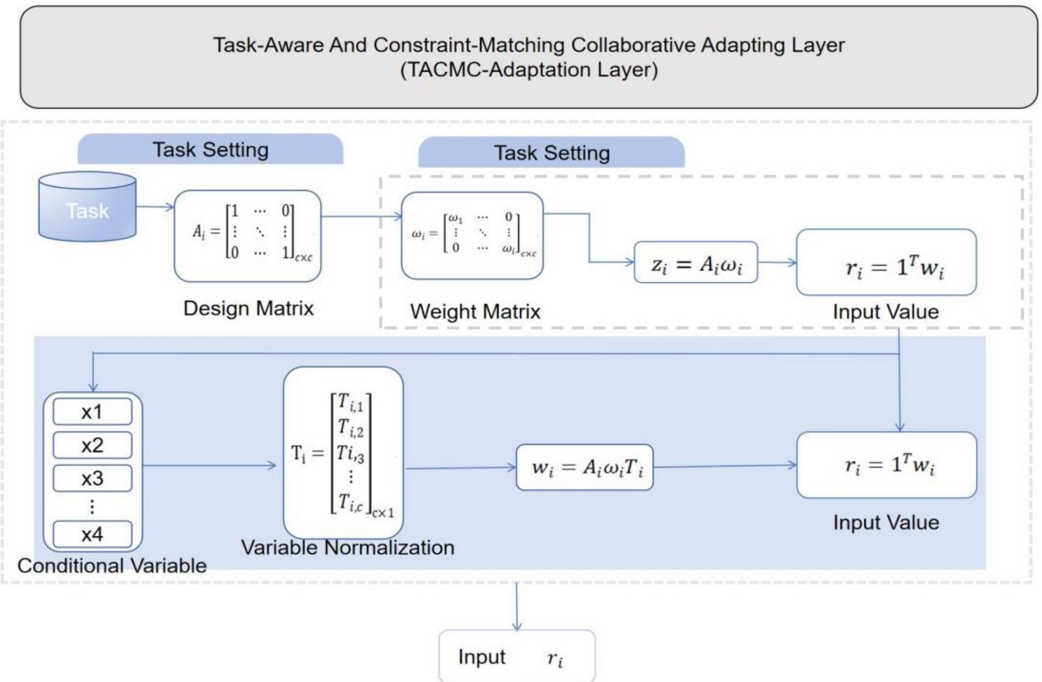


Figure 2. Multi-state heterogeneous scalar layer architecture diagram.

Dynamic adaptive distributed k-WTA network model

In practical warehouse environments, nodes are situated in highly dynamic, time-varying environments. Static topology communication may experience disconnection due to agent changes or environmental occlusion, thereby undermining overall system stability.

The AGV nodes in such spaces frequently adjust their positions and sensing states as handling tasks proceed. How to enable these nodes carrying standard inputs to quickly and stably complete distributed cooperative competition under conditions of communication constraints and noise interference is the core challenge in achieving reliable final task allocation.

To this end, this paper designs a dynamic topology adaptive k-WTA model to cope with frequent topology shifts in real-world warehouse scenarios.

Graph theory fundamentals for communication and dynamic topology definitions

A communication network composed of n AGVs can be represented by an undirected graph G=(V,E) Here, V={v_1,v_2,...,v_n} represents the vertex set of the graph, and assuming there are m edges, E={e_1,e_2,...,e_m} represents the edge set of the graph. An edge e_i in (v_i, v_j) the set is formed by a vertex pair meaning that a connecting edge is established between vertices v_i and v_j. The topological relationship of the entire graph can be described by an adjacency matrix A=$[a_{ij}]_{n\times n}$. When $(v_i, v_j) \in E$, the matrix element $a_{ij} > 0$; otherwise, $a_{ij} = 0$. If vertex v_i can be reached starting from vertex v_j, it is considered that a path exists between the two vertices.

Two vertices with a path between them are said to be connected. When any two vertices in graph G remain connected, graph G is a connected graph. For $v_j \in V$ and $(v_i, v_j) \in E$, it indicates that v_j is a neighbor of v_i. All neighboring vertices of vertex v_i constitute the neighbor set, denoted as $N(i)$.

For an undirected graph, the total number of neighbors possessed by vertex v_i is its degree. At this point, the degree matrix of the graph is expressed in the form of a diagonal matrix, written specifically as:

$$D = [d_{ij}]_{n\times n}, d_{(ii)} = sum_{(j\in N_i)} a_{(ij)} \tag{7}$$

Correspondingly, the Laplacian matrix of the undirected graph can be computed as:

$$L = D - A \tag{8}$$

Let the set of all possible static communication topologies be denoted as $P = \{1,2,\dots,M\}$ where each index $p \in P$ corresponds to an undirected graph $G_p = (V, E_p)$.

$V = \{1,\dots,N\}$ is the vertex set, and $E_p \subseteq V \times V$times V is the edge set under topology p. The adjacency matrix and Laplacian matrix of graph G_P are denoted as $A_p = [a_{ij}^p]$ and $L_p = L(G_p)$, respectively.

However, considering that communication links in practical applications may change due to factors such as distance variation, environmental interference, or node motion, this paper assumes that the system communication topology switches among a finite set of undirected graphs $\{G_1, G_2, \dots, G_M\}$, and the switching process is determined by a switching signal $\sigma(t)$ thereby forming a dynamic undirected topology $G_{\sigma(t)}$.

The variation of the communication topology over time is described by a switching signal:

$$\sigma(t): [0, +\infty) \to P \tag{9}$$

Where $\sigma(t) = p$ indicates that the communication topology of the system at time t is G_P.Thus, the actual communication graph of the upper-layer k-WTA network at time t can be written as:

$$G_p(t) = (V, E_p(t)) \tag{10}$$

Its adjacency matrix and Laplacian matrix are respectively denoted as:

$$A(t) = A_{\sigma(t)},\ L(G_{\sigma(t)}) = L_{\sigma(t)} \tag{11}$$

The switching signal $\sigma(t)$ is a piecewise constant function. There exists a sequence of increasing time instants$0 = t_0 \le t_1 \le t_2 \le \cdots$, such that on each time interval $[t_k, t_k + T)$,$\sigma(t)$ takes a constant value $\sigma_k \in P$, thereby keeping the topology $G_{\sigma(t)}$ unchanged within this interval, and switching occurs only at the discrete time instants t_k. The number of switches is finite in any finite time interval.

Under the topology $G_{\sigma(t)}$, the neighbor set of the i-th node is defined as

$$N_{\sigma(t)}(i) = \{j \in V | (i,j) \in E_{\sigma(t)}\} \tag{12}$$

The corresponding edge weight is denoted $a_{ij}(t) = a_{ij}^p$, $a_{ij}(t) = a_{ji}(t) \ge 0$, and when an edge exists, it has a uniform lower bound and upper bound, i.e.$aij\sigma(t)_{max_{min}}$, where $amax_{min}$are constants. This paper set the topology graph of transformation to be G_P, and now set its switching expression as:

$$G_p = \begin{cases} G_1 & 0 \le t < t_1 \\ G_2 & t_1 \le t < t_2 \\ G_3 & t_2 \le t < t_3 \\ G_4 & t_3 \le t < t_4 \end{cases} \tag{13}$$

As shown in Figure 3, the time-varying joint undirected network topology graph, where graphs (a)-(d) respectively represent the topology state graphs at four-time instants. The superposition of the edge sets of graphs (a)-(d) constitutes the joint connected total topology is a connected undirected graph.

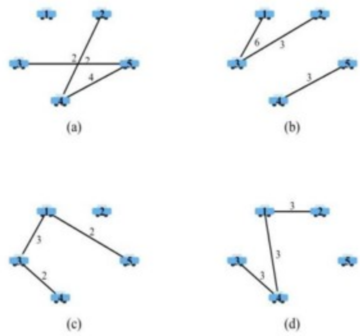


Figure 3. Time-varying joint undirected network topology graph.

Let there exists a constant $T > 0$ such that for any initial time $t_0 \ge 0$, the union graph of all communication graphs on the time window $t_0, t_0 + T$).

$$G([t_0, t_0 + T)) = \left(V, \cup_{t\in[t_0,t_0+T)} E_{\sigma(t)}\right) \tag{14}$$

This assumption relaxes the requirement that "the graph is connected at every instant," allowing local disconnections within a short time, as long as the nodes remain indirectly reachable among each other over a longer time scale. This is a more realistic topology model for mobile multi-agent networks. The topology can temporarily disconnect within a short time scale, but within a sufficiently long-time window, nodes remain indirectly reachable through a series of time-varying edges. This mechanism realizes the temporal propagation and global interaction of information through topology switching, enabling the distributed k-WTA network to still complete global competition and consistency optimization under dynamic communication environments.

Distributed k-WTA model under dynamic topologies

In this section, this paper first introduces the distributed k-WTA model in Literature. The distributed model in the literature is as follows:

$$\begin{cases} \varepsilon \dot{y}_i = -y_i + f_\Omega\left(\frac{1}{\alpha} r_i - \lambda_i\right) \\ \varepsilon \dot{z}_i = \frac{1}{\alpha}\sum_{j\in N_{\varphi(t)}} a_{ij}\left(\lambda_i - \lambda_j\right) \\ \varepsilon \dot{\lambda}_i = \frac{1}{\alpha}\left(y_i - \frac{k}{n} - \sum_{j\in N_{\varphi(t)}} a_{ij}\left(z_i - z_j\right) - \sum_{j\in N_{\varphi(t)}} a_{ij}\left(\lambda_i - \lambda_j\right)\right) \end{cases} \tag{15}$$

Where $\dot{y}_i = dy_i/dt$、$\dot{z}_i = dz_i/dt$、$\dot{\lambda}_i = d\lambda_i/dt$ are the first-order derivatives of each variable with respect to time, respectively; r_i is the input of the i-th AGV; y_i is the output of the i-th AGV.

If $y_i = 1$, it indicates that the corresponding input r_i is among the top-K largest values globally; otherwise $y_i = 0$; z_i and λ_i are two introduced state variables; a_{ij} is the element in the i-th row and j-th column of the adjacency matrix; α and ε are two positive constants.

The piecewise expression of the function $f_\Omega(\cdot)$ is as follows:

$$f_\Omega(y_i) = \begin{cases} 1 & y_i > 1, \\ 0 & y_i < 0, \\ y_i & 0 \le y_i \le 1. \end{cases} \tag{16}$$

As can be seen from Eq.(15), agent i only needs to exchange information with its neighbor j to complete the update of the agent's output y_i.When we introduce the multi-dimensional heterogeneous feature dimensionality reduction mapping model, the model becomes:

$$\begin{cases} \varepsilon \dot{y}_i = -y_i + f_\Omega\left(\frac{1}{\alpha} 1^T \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}_{c\times c} \begin{bmatrix} \omega_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \omega_c \end{bmatrix}_{c\times c} \begin{bmatrix} T_{i,1} \\ T_{i,2} \\ T_{i,3} \\ \vdots \\ T_{i,c} \end{bmatrix}_{c\times 1} - \lambda_i \right) \\ \varepsilon \dot{z}_i = \frac{1}{\alpha}\sum_{j\in N_{\varphi}(t)} a_{ij}\left(\lambda_i - \lambda_j\right) \\ \varepsilon \dot{\lambda}_i = \frac{1}{\alpha}\left(y_i - \frac{k}{n} - \sum_{j\in N_{\varphi}(t)} a_{ij}\left(z_i - z_j\right) - \sum_{j\in N_{\varphi}(t)} a_{ij}\left(\lambda_i - \lambda_j\right)\right) \end{cases} \tag{17}$$

Furthermore, the formula can be simplified as:

$$\begin{cases} \varepsilon \dot{y}_i = -y_i + f_\Omega\left(\frac{1}{\alpha} 1^T A_i \omega_i T_i - \lambda_i\right) \\ \varepsilon \dot{z}_i = \frac{1}{\alpha}\sum_{j\in N_{\varphi}(t)} a_{ij}\left(\lambda_i - \lambda_j\right) \\ \varepsilon \dot{\lambda}_i = \frac{1}{\alpha}\left(y_i - \frac{k}{n} - \sum_{j\in N_{\varphi}(t)} a_{ij}\left(z_i - z_j\right) - \sum_{j\in N_{\varphi}(t)} a_{ij}\left(\lambda_i - \lambda_j\right)\right) \end{cases} \tag{18}$$

Let $y = [y_1, y_2, \ldots, y_n]^T$, $A\omega T = [1^T A_1 \omega_1 T_1, 1^T A^2 \omega^2 T^2, \ldots, 1^T A^n \omega^n T^n]^T$, $z = [z_1, z_2, \ldots, z_n]^T$, $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_n]^T$. The original system can be rewritten as:

$$\begin{cases} \varepsilon\dot{y} = -y + f_{\Omega}\left(\frac{1}{\alpha}(A\omega T) - \lambda\right) \\ \varepsilon\dot{z} = \frac{L\lambda}{\alpha} \\ \varepsilon\dot{\lambda} = \frac{1}{\alpha}\left(y - \frac{k}{n}1 - Lz - L\lambda\right) \end{cases} \tag{19}$$

Under the aforementioned time-varying topology model, replacing the neighbor set $N(i)$ and edge weight a_{ij} with time-dependent $N_{\sigma(t)}(i)$ and $a_{ij}(t)$, yields the distributed k-WTA network model applicable to dynamic undirected graphs:

$$\begin{cases} \varepsilon\dot{y}_i(t) = -y_i(t) + f_{\Omega}\left(\begin{matrix}\frac{1}{\alpha}(1^T A_i\omega_i T_i(t) \\ -\lambda_i(t))\end{matrix}\right) \\ \varepsilon\dot{z}_i(t) = \frac{1}{\alpha}\sum_{j\in N_{\varphi(t)}} a_{ij}(t)(\lambda_i(t) - \lambda_j(t)) \\ \varepsilon\dot{\lambda}_i(t) = \frac{1}{\alpha}\left(\begin{matrix} y_i(t) - \frac{k}{n} - \sum_{j\in N_{\varphi(t)}} a_{ij}(t)(z_i(t) \\ -z_j(t)) \\ -\sum_{j\in N_{\varphi(t)}} a_{ij}(t)(\lambda_i(t) - \lambda_j(t)) \end{matrix}\right) \end{cases} \tag{20}$$

Similarly, the system can be simplified as:

$$\begin{cases} \varepsilon\dot{y} = -y(t) + f_{\Omega}\left(\frac{1}{\alpha}(AWT(t) - \lambda(t))\right) \\ \varepsilon\dot{z} = -\frac{1}{\alpha}L(G_{\sigma(t)})\lambda(t) \\ \varepsilon\dot{\lambda} = \frac{1}{\alpha}\left(\begin{matrix} y(t) - \frac{k}{N}[1,\ldots,1]^T \\ -L(G_{\sigma(t)})z(t) - L(G_{\sigma(t)})\lambda(t) \end{matrix}\right) \end{cases} \tag{21}$$

Figure 4 shows the transformation topology graph with 100 nodes, and Figures 5-8 show the system response graphs when selecting 5 winning nodes out of 100 nodes. Figure 5 is the evolutionary trajectory graph of the output state y with respect to time t. Figure 6 is the time response curve of the sum of output states of all agents, $sumy$.

Figure 7 shows that the system can find a new steady state when the topology switches. Figure 8 displays the optimization trajectory of the Lagrange multiplier λ, showing that after an initial rapid response, the λ of all 100 nodes perfectly converges to the same steady-state value.

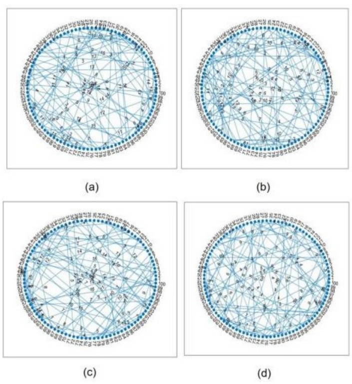


Figure 4. Transformation topology diagram of 100 nodes.

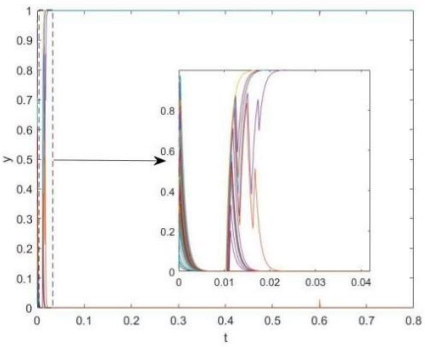


Figure 5. System state response curves.

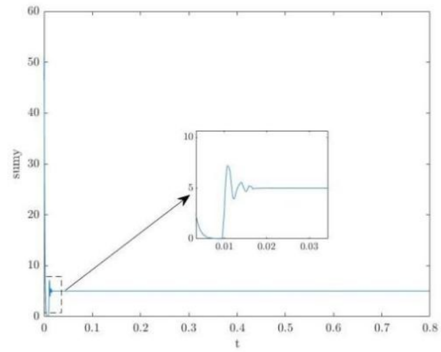


Figure 6. Evolution curves of the number of winning agents.

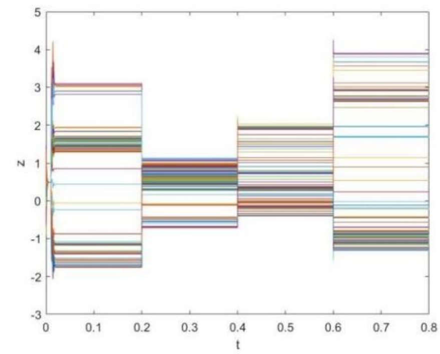


Figure 7. Transient behaviors of the state variables.

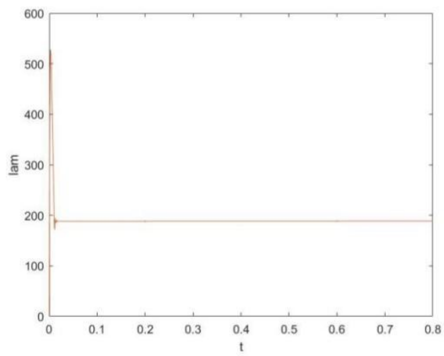


Figure 8. Response curves of the λ core constraint condition.

Activation functions directly determine the response characteristics of individual states to input information, affecting the convergence speed and steady-state accuracy of the network. Especially under dynamic topology environments, periodic changes in the network communication structure may lead to local bias in information reception, causing response lag in some nodes. Therefore, the activation function needs not only to possess rapid response capability, but also to balance stability during the network convergence process.

Traditional activation functions usually adopt fixed and unchangeable mathematical forms, making it difficult to simultaneously balance rapid response capability and smooth convergence performance in neural network training.

To tackle these inherent problems existing in conventional activation functions for neural network training, this study systematically analyzes the input-output response curve characteristics of typical activation functions used in deep learning models. On this basis, it innovatively introduces a flexible power adjustment mechanism together with the bounded nonlinear property of the inverse tangent function.

A novel power-adjusted activation function named the Power-arctangent activation function (Power-arctan) is therefore rationally constructed for network optimization. This newly designed function adaptively adjusts the response intensity in different input intervals and fully utilizes the unique bounded saturation feature of the inverse tangent function to limit excessively large output responses and suppress gradient explosion issues. Thereby achieving an effective and reliable trade-off between rapid convergence speed and superior steady-state stability in model training. Its specific mathematical expression is scientifically defined as follows:

$$F(x)=\begin{cases}\frac{arctan(\gamma x)}{arctan(\gamma)} & |x|<1\\ \mathrm{sgn}(x)\cdot|x|^{\tau} & |x|\geq 1\end{cases} \tag{22}$$

where， $\tau>1$ is the external acceleration gain hyper parameter, and $\gamma>0$ is the internal shape adjustment hyperparameter. The function is divided into two response regions based on the amplitude of the input state: When $|x|<1$ a normalized arctangent mapping is adopted; when $|x|\geq 1$ a signed power mapping is adopted.

Small-deviation region arctangent response

When the system state gradually approaches the target state, $|x|<1$. At this time, it adopts:

$$F(x)=\frac{arctan(\gamma x)}{arctan(\gamma)} \tag{23}$$

The arctangent function possesses the characteristics of being continuous, monotonic, and bounded, thereby avoiding excessively drastic nonlinear responses when approaching the equilibrium state. Meanwhile, through the parameter γ, the response curvature in this region can be adjusted, thereby changing the adjustment characteristics of the system under small deviation conditions.

In addition, at the piecewise boundary $x=\pm 1$:

$$F(1^-)=\frac{arctan(\gamma)}{arctan(\gamma)}=1 \tag{24}$$

And:

$$F(1^+)=1^{\tau}=1 \tag{25}$$

Similarly:

$$F(-1^-)=F(-1^+)=-1 \tag{26}$$

Therefore, the activation function has function value continuity at the piecewise boundary, which can avoid extra state disturbances caused by sudden changes in function values.

Power response in the large-deviation region

When the system is affected by topology switching and produces a large state deviation, i.e., $|x|\geq 1$, the activation function adopts:

$$F(x)=\mathrm{sgn}(x)\cdot|x|^{\tau} \tag{27}$$

At this time, the parameter τ determines the response intensity in the large-deviation region. When $\tau>1$, the activation function exhibits superlinear growth characteristics, which can enhance the state adjustment capability of the system under large deviations; when $0<\tau<1$, it can decelerate the response growth rate.

Therefore, by rationally selecting, τ the large-deviation response capability of the system can be adjusted according to specific network scales and dynamic topology variation characteristics. The K-WTA function at this time is:

$$\begin{cases}\varepsilon\dot{y}=-y(t)+F\left(\frac{1}{\alpha}(AWT(t)-\lambda(t))\right)\\ \varepsilon\dot{z}=-\frac{1}{\alpha}L(G_{\sigma(t)})\lambda(t)\\ \varepsilon\dot{\lambda}=\frac{1}{\alpha}\begin{pmatrix}y(t)-\frac{k}{N}[1,\ldots,1]^T\\ -L(G_{\sigma(t)})z(t)-L(G_{\sigma(t)})\lambda(t)\end{pmatrix}\end{cases} \tag{28}$$

To test the noise-filtering performance of the power-arctan function, constant noise and random noise are set for comparison with the agent signals generated without noise.

Figure 9 is the waveform in a noise-free environment, Figure 10 is the waveform under constant noise $\theta=$

5, $\gamma = 4$, $\tau = 5$ and Figure 11 is the waveform under random noise $\theta \in (3,\ 6)$, $\gamma = 8$, $\tau = 5$.

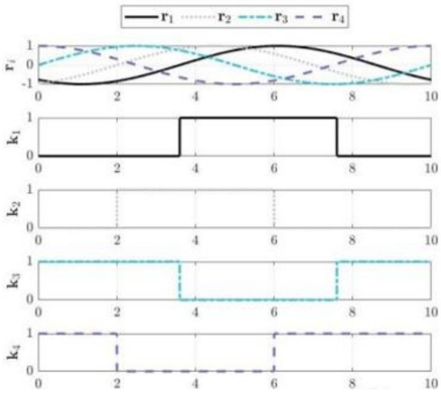


Figure 9. Noise-free waveform curves.

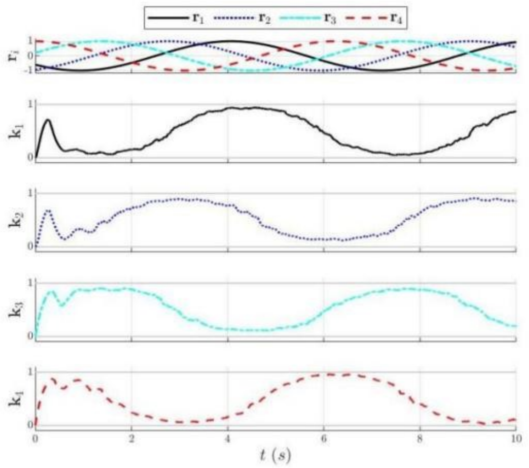


Figure 10. Constant-noise waveform curves.

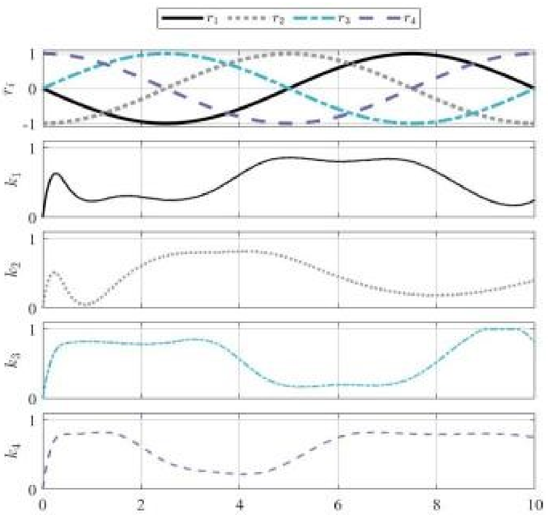


Figure 11. Random-noise waveform curves.

Adaptive gain strategy

In traditional fixed-gain models, the control gain parameter is often set as a fixed constant throughout the entire operation process. However, if an extremely high gain value is manually set in advance, although the system can achieve an ultrafast response speed in the initial stage of iteration, it will produce severe numerical oscillations and unstable fluctuations when dealing with large-scale swarms or extremely large initial errors. Conversely, if an extremely low gain is adopted, the convergence process of the system will be extremely sluggish and time-consuming, failing to meet the strict real-time requirements of complex practical engineering scenarios.

To overcome the inherent contradictions and performance limitations brought by the aforementioned fixed gain parameters that cannot adapt to dynamic system states, this paper proposes an innovative adaptive dynamic gain strategy (proposed adaptive gain strategy). The core idea of this effective strategy is to dynamically adjust the real-time gain magnitude according to the current global convergence error of the system throughout the whole iteration process, so as to balance the system convergence speed and operational stability in real time. The global competitive tracking error of the system at time is defined as:

$$e(t) = |\textstyle\sum_{i=1}^{n} x_i\,(t) - K| = \|1^T y(t) - K\| \tag{29}$$

Based on the aforementioned error feedback, the specific expression of the dynamic adaptive control gain $\varepsilon(t)$ designed in this paper is as follows:

$$\varepsilon(t) = \varepsilon\left(1 - e^{-\frac{1}{2}e(t)^2}\right) minmax_{min} \tag{30}$$

Where: ε_{max} is the maximum allowable upper bound of the gain set by the system, used to provide the maximum energy control under extreme conditions. ε_{min} is the lower bound of the gain to ensure the basic convergence capability and steady-state accuracy of the system.

To further verify the adaptability and robustness of the proposed adaptive gain strategy, we focus on complex dynamic communication topology changes in practical AGV swarm operation scenarios. For this purpose, a typical scenario with periodic communication topology switching is adopted to carry out simulation verification. In the experimental setup, four different network scales with n=30, 50, 100, 200 are adopted to simulate swarm systems of different sizes, while the target selection number is fixed at k=5 for all comparative test groups to ensure consistent experimental variables.

Figure 12 intuitively presents the detailed convergence comparison results between the optimized adaptive gain method and the traditional fixed gain method under diverse network scales.

The time taken for the system to re-reach the preset convergence criterion after each topology switch is defined as the topology recovery time $\overline{T_r}$. To avoid the influence of initial states on the experimental results, the recovery times after three topology switches are statistically averaged. The experimental results are shown in Table 1.

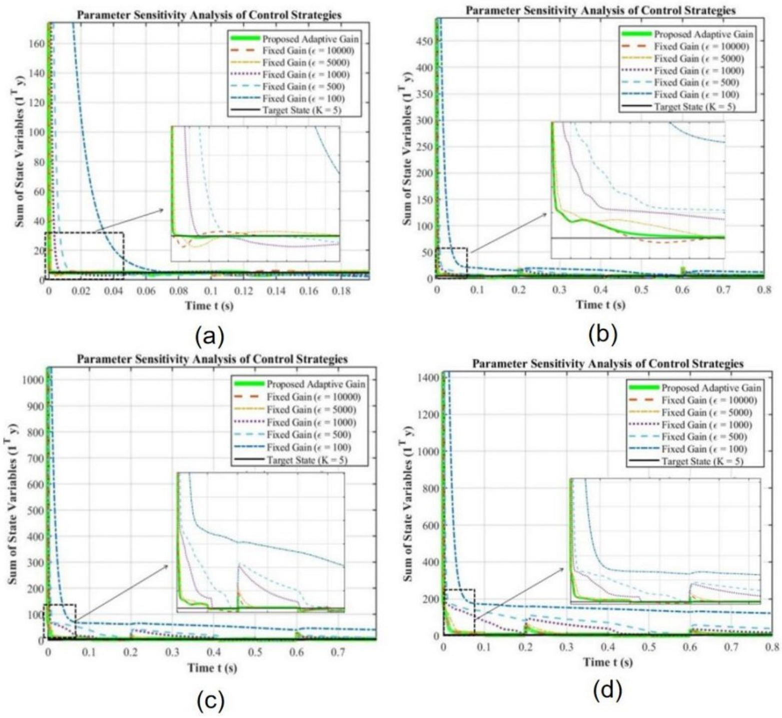


Figure 12. Convergence comparison between adaptive gain and fixed gain under different network scales. (a) is the gain convergence comparison for n=30, (b) is the gain convergence comparison for n=50, (c) is the gain convergence comparison for n=100, and (d) is the gain convergence comparison for n=200.

Table 1. Recovery times after topology switching under different network scales.

$\varepsilon \setminus n$	30	50	100	200
100	0.1730 s	NC	NC	NC
500	0.1477 s	0.2073 s	0.3079 s	0.3666 s
1,000	0.1362 s	0.1827 s	0.2044 s	0.2428 s
5,000	0.0196 s	0.0997 s	0.1072 s	0.1374 s
10,000	0.0064 s	0.0252 s	0.0909 s	0.1164 s
Adaptive	0.0036 s	0.0176 s	0.0676 s	0.0857 s

Note: NC= Not converged within the prescribed simulation time.

As can be seen from the experimental figures, high gain can suppress the state sum at an extremely fast speed; however, observing the locally enlarged regions of each subfigure reveals that when approaching the target state, excessively large control gains induce obvious undershoot phenomena, destroying the local stability of the system. This indicates that fixed gain is difficult to simultaneously balance the rapid convergence requirement in the large-error stage and the dynamic stability requirement in the small-error stage.

In contrast, the proposed adaptive gain strategy can recover to a steady state within a relatively short time under different network scales, with average recovery times of 0.0036 s, 0.0176 s, 0.0676 s, and 0.0857 s, respectively.

Especially in large-scale networks with n=100 and n=200, the recovery time of the adaptive gain is reduced by approximately 25.63% and 26.37% compared to the fixed gain$\varepsilon = 10000$, respectively. This indicates that the proposed method can dynamically adjust the control gain according to the system state after topology switching, thereby accelerating the recovery process of the network from the topology-perturbed state to the steady state. Therefore, the proposed adaptive gain strategy exhibits faster recovery speed and better network scale scalability under dynamic topology environments.

Experimental verification

In practical intelligent logistics and automated truck loading scenarios, the AGV fleet scale is often relatively large, and unexpected situations such as single-vehicle equipment failure or battery depletion inevitably occur during task execution. To verify the high efficiency of the proposed K-WTA-based task allocation model under large-scale fleets and its dynamic self-healing capability against sudden failures, a multi-AGV cooperative handling and task allocation experiment with n=100 was designed. The experiment constructed a 50×50 two-dimension- al grid-based storage area. The initial positions, current battery levels, and working statuses (Idle/Busy) of the 100 AGVs are randomly initialized and generated by the system.

A core loading task point (represented by the red pentagram in the figure) is set in the center of the environment. The system sets the optimal number of AGVs required for a single cooperative handling task to $k = 5$. The system noise is a constant value of $\theta = 2$, the maximum driving speed of the AGVs is set to v_{max} and the acceleration range during operation is strictly limited between [-1.5, 1.0] m/s^2. In the TACMC-adaptation layer:

$$T_1 = exp(-0.05\|v_i(t) - v_0(t)\|) \tag{29}$$

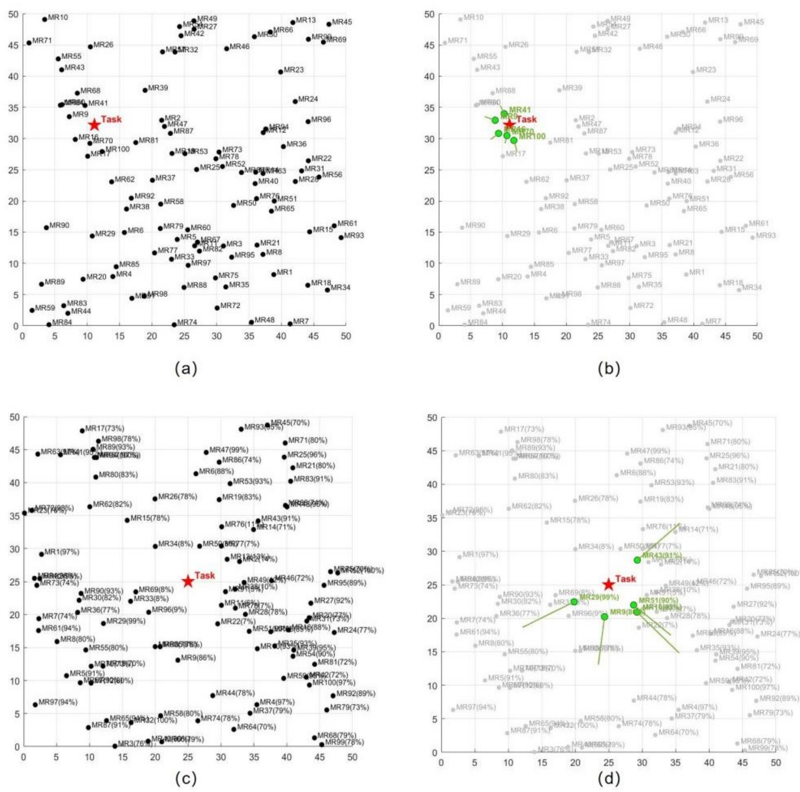
Where $T \in (0,1]$.The closer the distance, the closer T_1 is to 1.

$$T_2 = \frac{E_i(t)}{E_{\max}} \tag{30}$$

The more sufficient the battery level, the larger T_2, indicating a stronger capability to continuously execute tasks.

$$T_3 = \begin{cases} 1, & Idle \\ 0, & Busy \end{cases} \tag{31}$$

Figures 13 (a) and (b) show that the system uses a single distance-based metric, i.e.,$r_i = -\|v_i(t) - v_0(t)\|$, to select MR9, MR16, MR41, MR70, and MR100 as winning AGVs. Figures 13 (c) and (d) show that the system uses two metrics - distance and battery level - i.e.$r_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.8 & 0 & 0 \\ 0 & 0.2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix}$, to select MR9, MR10, MR29, MR43, and MR51 as winning AGVs. Figures 13 (e) and (f) show that the system uses three indicators - distance, battery level, and workload status - i.e.,$r_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.8 & 0 & 0 \\ 0 & 0.2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix}$, to select MR76, MR79, MR62, MR48, and MR39.



(a) (b)

(c) (d)

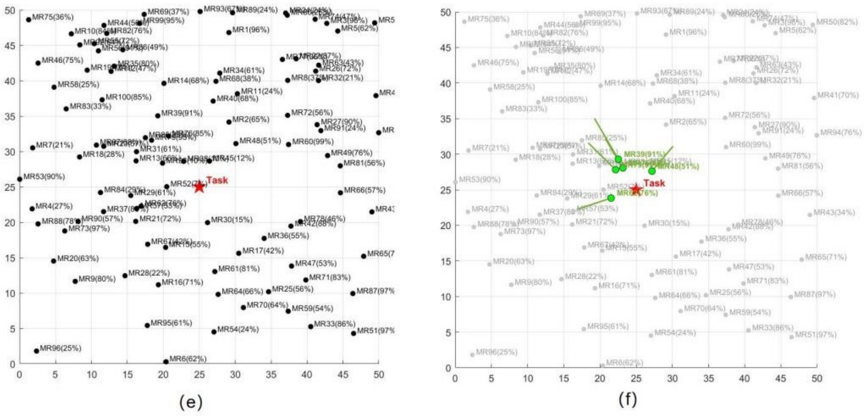


Figure 13. Multi-state heterogeneous AGV experimental diagrams.

Conclusion

To address the task allocation problem in multi-AGV automated truck loading systems under dynamic communication environments, this paper proposes a robust adaptive distributed K-WTA task allocation method under time-varying topologies. The method integrates heterogeneous features to represent task information and utilizes time-varying topologies to enhance node collaboration.

A power-arctan activation function and an adaptive gain strategy are adopted to dynamically adjust the network response. Simulation results confirm that the method possesses certain anti-noise capabilities, and its recovery speed after topology switching outperforms that of fixed-gain schemes.

Parameter analysis provides a basis for parameter tuning, and the truck-loading scenario simulation verifies the effectiveness of task allocation, offering a viable scheme for distributed task allocation of multiple AGVs under dynamic communication. Future work will consider realistic factors such as communication delays and packet loss and will conduct large-scale cluster simulations and AGV physical experiments to examine the engineering application effectiveness of the method.

Funding

This work was supported by the Key R&D Program of Shandong Province, China under Grant Nos: (2024CXGC010809, 2025TSGCCZZB0597, 2025TSG CCZZB0365, 2025TSGCCZZB0390, 2025 TSGCCZZB 0369 and 2025TSGCCZZB0756).

Acknowledgements

The authors would like to show sincere thanks to those technicians who have contributed to this research.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] Nguyen, T. P., Nguyen, H., Phan, D. M., Ngo, H. Q. T. (2026) Dynamic chain for scheduling of the multi-AGV systems with load-aware motion profiling. *Scientific Reports*, 16(1), 6832.

[2] Wang, X., Wu, D., Wang, X., Zeng, R., Ma, L., Yu, R. (2023) Truthful auction-based resource allocation mechanisms with flexible task offloading in mobile edge computing. *IEEE Transactions on Mobile Computing*, 23(5), 6377-6391.

[3] Zhang, L., Xiao, K., Jin, L., Dong, P., Tong, Z. (2023) Mobility-aware and double auction-based joint task offloading and resource allocation algorithm in MEC. *IEEE Transactions on Network and Service Management*, 21(1), 821-837.

[4] Choi, H. L., Brunet, L., How, J. P. (2009) Consensus-based decentralized auctions for robust task allocation. *IEEE Transactions on Robotics*, 25(4), 912-926.

[5] Sun, Y., Dong, K. (2025) A refined model for check-in and security staff allocation at airports based on an improved bayesian neural network. *IEEE Access*, 13, 47230-47243.

[6] Zhen, Z., Wen, L., Wang, B., Hu, Z., Zhang, D. (2021) Improved contract network protocol algorithm based cooperative target allocation of heterogeneous UAV swarm. *Aerospace Science and Technology*, 119, 107054.

[7] Chen, C., Bao, W., Men, T., Zhu, X., Wang, J., Wang, R. (2020) NECTAR-an agent-based dynamic task allocation algorithm in the UAV swarm. *Complexity*, 2020(1), 6747985.

[8] Qu, H., Seifan, T., Tielbörger, K., Seifan, M. (2013) A spatially explicit agent-based simulation platform for investigating effects of shared pollination service on ecological communities. *Simulation Modelling Practice and Theory*, 37, 107-124.

[9] Zhang, M., Chen, H., Cai, W. (2023) Hunting task allocation for heterogeneous multi-AUV formation target hunting in IoUT: a game theoretic approach. *IEEE Internet of Things Journal*, 11(5), 9142-9152.

[10] Huo, X., Zhang, H., Huang, C., Wang, Z., Yan, H. (2023) Task allocation with minimum requirements for multiple mobile robot systems: a game-theoretical approach. *IEEE Transactions on Network Science and Engineering*, 11(1), 1202-1213.

[11] Xing, Y., Chen, C., Wu, J., Chen, J. (2025) Reinforcement learning for efficient multiagent task allocation in potential game model. *IEEE Transactions on Artificial Intelligence*, 6(12), 3217-3231.

[12] Jang, I., Shin, H. S., Tsourdos, A. (2018) Anonymous hedonic game for task allocation in a large-scale multiple agent system. *IEEE Transactions on Robotics*, 34(6), 1534-1548.

[13] Wu, H., Shang, H. (2020) Potential game for dynamic task allocation in multi-agent system. *ISA Transactions*, 102, 208-220.

[14] Wu, J., Zhang, J., Sun, Y. N., Li, X., Gao, L., Han, G. (2023) Multi-UAV collaborative dynamic task allocation method based on ISOM and attention mechanism. *IEEE Transactions on Vehicular Technology*, 73(5), 6225-6235.

[15] Liu, Y., Song, R., Bucknall, R., Zhang, X. (2019) Intelligent multi-task allocation and planning for multiple unmanned surface vehicles (USVs) using self-organising maps and fast marching method. *Information Sciences*, 496, 180-197.

[16] Ma, S., Guo, W., Song, R., Liu, Y. (2021) Unsupervised learning based coordinated multi-task allocation for unmanned surface vehicles. *Neurocomputing*, 420, 227-245.

[17] Shima, T., Rasmussen, S. J., Sparks, A. G., Passino, K. M. (2006) Multiple task assignments for cooperating uninhabited aerial vehicles using genetic algorithms. *Computers & Operations Research*, 33(11), 3252-3269.

[18] Jiang, C., Wan, Z., Peng, Z. (2020) A new efficient hybrid algorithm for large scale multiple traveling salesman problems. *Expert Systems with Applications*, 139, 112867.

[19] Chen, Z., Alonso-Mora, J., Bai, X., Harabor, D. D., Stuckey, P. J. (2021) Integrated task assignment and path planning for capacitated multi-agent pickup and delivery. *IEEE Robotics and Automation Letters*, 6(3), 5816-5823.

[20] Luo, L., Chakraborty, N., Sycara, K. (2015) Distributed algorithms for multirobot task assignment with task deadline constraints. *IEEE Transactions on Automation Science and Engineering*, 12(3), 876-888.

[21] Mazdin, P., Rinner, B. (2021) Distributed and communication-aware coalition formation and task assignment in multi-robot systems. *IEEE Access*, 9, 35088-35100.

[22] Yu, J., Chung, S. J., Voulgaris, P. G. (2014) Target assignment in robotic networks: distance optimality guarantees and hierarchical strategies. *IEEE Transactions on Automatic Control*, 60(2), 327-341.

[23] Zhao, W., Meng, Q., Chung, P. W. (2015) A heuristic distributed task allocation method for multivehicle multitask problems and its application to search and rescue scenario. *IEEE Transactions on Cybernetics*, 46(4), 902-915.

[24] Zhang, Y., Parker, L. E. (2012) IQ-ASyMTRe: forming executable coalitions for tightly coupled multirobot tasks. *IEEE Transactions on Robotics*, 29(2), 400-416.

[25] Vig, L., Adams, J. A. (2006) Multi-robot coalition formation. *IEEE Transactions on Robotics*, 22(4), 637-649.

[26] Rizk, Y., Awad, M., Tunstel, E. W. (2019) Cooperative heterogeneous multi-robot systems: a survey. *ACM Computing Surveys (CSUR)*, 52(2), 1-31.

[27] Liu, S., Wang, J. (2006) A simplified dual neural network for quadratic programming with its k-WTA application. *IEEE Transactions on Neural Networks*, 17(6), 1500-1510.

[28] Qi, Y., Jin, L., Luo, X., Shi, Y., Liu, M. (2021) Robust k-WTA network generation, analysis, and applications to multiagent coordination. *IEEE Transactions on Cybernetics*, 52(8), 8515-8527.

[29] Liu, M., Zhang, X., Shang, M., Jin, L. (2022) Gradient-based differential k-WTA network with

application to competitive coordination of multiple robots. *IEEE/CAA Journal of Automatica Sinica*, 9(8), 1452-1463.

[30] Zhang, Y., Li, S., Weng, J. (2022) Distributed k-winners-take-all network: an optimization perspective. *IEEE Transactions on Cybernetics*, 53(8), 5069-5081.

[31] Liu, K., Zhang, Y. (2024) Distributed dynamic task allocation for moving target tracking of networked mobile robots using k-WTA network. *IEEE Transactions on Neural Networks and Learning Systems*, 36(3), 5795-5802.

[32] Liu, M., Ding, J., Chen, Y., Jin, L. (2025) Time-delayed k-WTA network considering communication interference with multirobot applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 55(12), 9889-9899.