A Multi-Agent Collaboration-Based Edge Data Processing and Decision-Making Mechanism for the Industrial Internet of Things

Chunqi Jiao, Xuan Liu*

Harbin Institute of Information Technology, Harbin, 150431, China
**Corresponding author:m15174507563@163.com*

Abstract: *The massive amount of data generated at the edge side of the Industrial Internet of Things imposes stringent requirements on processing real-time performance and decision-making intelligence, while traditional centralized architectures face latency and bandwidth bottlenecks. To address this challenge, a distributed processing framework based on multi-agent collaboration is constructed. Edge nodes are abstracted as autonomous decision-making entities through multi-agent modeling, a distributed consensus mechanism is adopted to achieve heterogeneous data synchronization, and an event-driven protocol is designed to optimize communication efficiency. A deep reinforcement learning model is introduced to enable online extraction of data features and dynamic task scheduling, and it is adapted to resource constraints through model lightweighting techniques. Furthermore, game theory and policy gradient methods are integrated to establish a collaborative decision-making mechanism in dynamic environments, ensuring the consistency of local strategies with global objectives. This study has constructed a complete technical system ranging from data interaction to intelligent decision-making, providing theoretical support for edge intelligence in the Industrial Internet of Things.*

Keywords: *Multi-Agent Collaboration; Industrial Internet of Things; Edge Computing; Deep Reinforcement Learning; Distributed Decision-Making; Game Theory*

Introduction

The edge side of the Industrial Internet of Things continuously generates high-throughput, multi-modal real-time data streams, the processing efficiency of which is directly critical to key applications such as production process monitoring and predictive maintenance. However, the industrial environment is complex and dynamic, with resource-constrained edge nodes and fluctuating network conditions, making it difficult for traditional centralized architectures to meet the requirements for millisecond-level response and high reliability. Although sinking computing to the edge alleviates transmission pressure, the isolated operational mode among nodes still fails to unlock the collaborative potential of distributed resources. Multi-agent system theory provides a naturally suitable modeling tool for distributed collaborative problem-solving, enabling the formation of collective intelligence through information sharing and policy coordination. Meanwhile, deep reinforcement learning has emerged as an ideal choice for handling dynamic decision-making scenarios due to its online learning and adaptive capabilities. Addressing the challenges introduced by multi-agent integration—such as interaction complexity, resource constraints in model deployment, and the maintenance of decision consistency in dynamic environments—this study intends to integrate multi-agent collaboration, deep reinforcement learning, and game theory. It aims to construct a systematic solution from three levels: edge computing architecture, distributed data processing models, and dynamic decision-making optimization mechanisms, thereby providing theoretical support for the intelligent evolution of the Industrial Internet of Things.

1. Multi-Agent Collaborative Edge Computing Architecture and Data Interaction Mechanisms

1.1 Multi-Agent System Modeling for the Industrial Internet of Things Edge

Edge nodes in an Industrial Internet of Things environment exhibit significant characteristics such as geographical dispersion, resource heterogeneity, and dynamic access, making it difficult for traditional centralized processing architectures to meet the requirements for low-latency response and high reliability. Multi-agent system theory provides a distributed modeling framework for edge computing, abstracting each edge node equipped with computing and communication capabilities as an agent possessing autonomous perception, decision-making, and execution abilities. These agents are endowed with local environmental perception capabilities, enabling them to perform independent reasoning based on limited information, while simultaneously forming collective intelligence through interaction mechanisms to collaboratively accomplish complex edge data processing tasks. During the modeling process, it is necessary to define the functional boundaries and action spaces of the agents, specify their internal state update rules, and establish interfaces for interaction with the external environment, thereby constructing a formal model adapted to the dynamic nature of industrial scenarios.

The core of model construction lies in characterizing the inter-agent relationships and the degree of task coupling. Considering the diversity of device types and varying data flow directions within the Industrial Internet of Things, the topological structure of the multi-agent system needs to reflect the logical connections and data dependencies among physical devices. By introducing graph theory methods, agents and their interaction relationships can be represented as a dynamic graph structure, where node attributes characterize computing and storage resource capabilities, and edge weights reflect communication quality and data flow characteristics. On this basis, the state space and action space of the agents are established, and reward functions tailored for specific industrial application scenarios are defined. This enables each agent, while pursuing local objectives, to simultaneously serve the system's overall optimization goals, thereby laying the model foundation for subsequent collaborative data processing[1].

1.2 Heterogeneous Data Aggregation and Synchronization Mechanism Based on Distributed Consensus

Data sources at the edge side of the Industrial Internet of Things exhibit multi-modal, high-dimensional, and non-aligned characteristics. Significant differences exist in the sampling frequencies, time-stamp references, and numerical scales of data generated by different sensors. Directly fusing them would lead to information distortion and decision deviation. The distributed consensus mechanism provides an effective approach to addressing this challenge. Through multiple rounds of information exchange and state negotiation among agents, a unified understanding of data semantics and time references can be achieved without coordination by a central node. In specific implementation, agents preprocess locally collected data, extract key features accompanied by timestamp information, and then exchange feature vectors with other agents through iterative interaction protocols, progressively eliminating ambiguity and redundancy among the data.

The construction of the synchronization mechanism requires balancing convergence speed and communication overhead. Constrained by limited bandwidth and fluctuating channel quality in industrial field networks, traditional fully connected consensus algorithms may lead to excessive communication load. By adopting a trust-based weighted consensus strategy, agents dynamically adjust the weights of neighbor node information based on the reliability of historical interactions, enabling the aggregation process to focus more on high-confidence data sources. Simultaneously, a logical clock mechanism is introduced to mark each data unit with a globally consistent temporal identifier, thereby resolving data out-of-order issues caused by transmission delays. Through the aforementioned mechanisms, the multi-agent system can form a unified view of the industrial process state at the edge, providing a consistent and reliable data foundation for higher-level decision-making.

1.3 Inter-Agent Communication Protocol and Load Balancing Strategy for Collaborative Processing

The collaborative efficiency of a multi-agent system is highly dependent on the efficiency of information exchange among agents. However, the constrained network resources and dynamic topological changes in the Industrial Internet of Things environment pose severe challenges to the design of communication protocols. Traditional periodic broadcasting or centralized aggregation models struggle to meet the real-time requirements of edge computing, necessitating the design of lightweight and adaptive interaction protocols. The protocol design adopts an event-triggered communication mechanism, where agents initiate information exchange only upon detecting significant changes in the environmental state or when their decision-making confidence falls below a threshold, thereby substantially reducing redundant communication traffic. In terms of packet structure, compressed sensing techniques are employed to perform dimensionality reduction encoding on the transmitted state information, reducing network load while preserving the integrity of critical information.

The formulation of the load balancing strategy needs to comprehensively consider the computational capacity, residual energy, and current task queue length of the agents. In a distributed environment, relying solely on local information for scheduling can easily lead to an imbalance where some nodes are overloaded while others remain idle. By introducing market mechanisms or game theory models, agents can decompose their own computational tasks and dynamically adjust task allocation schemes based on the resource bids of neighboring nodes. For control tasks with high real-time requirements, a priority-based preemptive scheduling strategy is adopted to ensure that critical data flows obtain sufficient computational resources. The collaborative optimization of the communication protocol and load balancing enables the multi-agent system to maintain efficient and stable operation within resource-constrained edge environments, thereby providing reliable support for complex industrial applications.

2. A Distributed Processing Model for Edge Data Based on Deep Reinforcement Learning

2.1 Online Extraction of Data Features and Methods for Multi-Dimensional Information Fusion

Data streams generated at the edge side of the Industrial Internet of Things exhibit characteristics of high throughput, non-stationarity, and strong correlation. Traditional offline feature extraction methods based on batch processing cannot meet the demands of real-time processing. An online feature extraction mechanism requires completing the analysis and compression of raw signals instantaneously upon data arrival, extracting essential attributes that can represent equipment operating status or environmental changes. By adopting a technical path that combines sliding windows and incremental learning, agents maintain dynamically updated feature buffers locally, performing Fast Fourier Transform and wavelet packet decomposition on time-series data within the window to extract multi-dimensional feature indicators in both the time and frequency domains. For unstructured data such as vibration signals or infrared thermography, a lightweight convolutional autoencoder structure is introduced to accomplish automatic feature learning and dimensionality reduction representation at the edge[2].

Multi-dimensional information fusion occurs at both the feature level and the decision level. At the feature level, the heterogeneous feature vectors collected by different agents suffer from inconsistent dimensions and semantic gaps, necessitating their transformation into a unified representation space through feature alignment and mapping. An attention mechanism is employed to dynamically evaluate the contribution of each dimensional feature to the current processing task, assigning higher fusion weights to critical features. At the decision level, preliminary judgments generated by each agent based on local fusion results require collaborative verification. Anomalous deviations caused by sensor noise or local faults are eliminated through Bayesian inference or evidence theory methods. This layered fusion architecture not only preserves the rich information of the raw data but also reduces the complexity of transmission and computation, thereby providing high-quality input for subsequent intelligent decision-making.

2.2 Dynamic Scheduling Mechanism for Distributed Processing Tasks Based on Deep Q-Networks

The diversity of data processing tasks in an edge environment and the bursty nature of their arrival patterns make it difficult for static task scheduling strategies to maintain the long-term effectiveness of the system. Deep Q-Networks provide a model-free reinforcement learning framework for learning optimal scheduling policies, enabling agents to autonomously acquire knowledge of task allocation and resource scheduling through continuous interaction with the environment. In the modeling process, the state of each agent is defined by its current task queue length, remaining computational resources, and the load information of neighboring nodes, while the action space includes various scheduling choices such as local execution, task migration, or collaborative processing. By constructing a deep neural network to approximate the state-action value function, agents can select scheduling actions that maximize cumulative rewards based on current environmental perceptions.

The dynamism of the scheduling mechanism is reflected in its ability to respond in real-time to changes in industrial site operating conditions. When a sudden data surge or edge node failure is detected, traditional rule-based scheduling systems often require manual intervention or reconfiguration.

In contrast, agents driven by Deep Q-Networks can automatically adjust their strategies by leveraging experience accumulated during past training. An experience replay mechanism is introduced to break the temporal correlations among training data, enabling agents to learn how to handle various abnormal scenarios from historical scheduling cases. Simultaneously, a dual-network structure is adopted to mitigate the risk of overestimating the value function, thereby enhancing the stability of scheduling decisions. Through continuous policy iteration, the multi-agent system gradually forms a distributed scheduling model adapted to complex industrial environments, achieving maximized global resource utilization while meeting task deadline requirements[3].

2.3 Model Lightweighting and Inference Acceleration Strategies under Resource Constraints of Edge Nodes

An inherent contradiction exists between the computationally intensive nature of deep reinforcement learning models and the limited resources of edge nodes, necessitating targeted lightweight transformations during the model deployment phase. Knowledge distillation technology provides an effective approach to solving this problem. By training a streamlined student network to simulate the behavioral patterns of a complex teacher network, the core decision-making capabilities of the Deep Q-Network are compressed to a scale suitable for edge execution. During the distillation process, the focus is not only on fitting the final decision outcomes but, more importantly, on retaining the teacher network's generalization ability across different regions of the state space, enabling the student network to still make reasonable judgments when faced with unseen industrial scenarios.

The implementation of inference acceleration strategies requires consideration of the synergistic optimization of hardware characteristics and algorithm structure. For the ARM architectures or FPGA platforms commonly used in edge devices, operator fusion and quantization are performed on the convolutional and fully connected layers within the network, converting 32-bit floating-point parameters into 8-bit fixed-point representations, thereby significantly reducing memory usage and computational latency. For processing tasks with strong temporal dependencies, a state caching mechanism for recurrent neural networks is introduced to avoid redundant computation of historical information. After model deployment, a continuous adaptive adjustment mechanism is established to dynamically switch between inference modes with different precision based on the real-time load of edge nodes: activating the lightweight version to ensure basic functionality during resource constraints, and reverting to the full model to enhance decision quality when resources are abundant. This series of optimization measures enables complex deep reinforcement learning models to operate efficiently within resource-constrained edge environments, providing reliable algorithmic support for intelligent data processing in the Industrial Internet of Things.

3. Collaborative Decision-Making and Strategy Optimization Mechanisms for Dynamic Environments

3.1 Multi-Agent Collaborative Decision-Making Modeling and Solution Based on Game Theory

The dynamic nature and uncertainty of the edge environment in the Industrial Internet of Things cause the decision-making interactions among multiple agents to exhibit complex strategic dependencies. Traditional optimization methods struggle to characterize the conflicts and coordination between individual rationality and collective interests. Game theory provides a systematic mathematical framework for analyzing such distributed decision-making problems. By modeling each agent's decision-making process as a game participant, its strategy space corresponds to possible action choices, and the payoff function reflects the contribution of the decision outcome to the quality of task completion. In collaborative data processing scenarios, the objectives among agents often contain both cooperative elements (e.g., jointly maintaining system stability) and competitive factors (e.g., competing for limited computational and communication resources). Therefore, employing a hybrid game model can more accurately describe this complex relationship. Potential game theory, due to its property of possessing pure-strategy Nash equilibria and the ease of convergence for distributed learning algorithms, becomes an ideal tool for modeling collaborative decision-making in edge intelligence. Its potential function can be directly linked to the measurement indicators of global data processing performance.

Solving the game model requires obtaining an equilibrium strategy while ensuring real-time performance. Due to the vast state space of the industrial environment and the fact that the payoff functions are not explicitly known, traditional analytical solution methods are difficult to apply. The combination of distributed reinforcement learning and game theory provides a feasible path for this: agents iteratively update their strategies using algorithms such as fictitious play or gradient ascent by observing their own historical decisions and obtained payoffs, gradually approaching a Nash equilibrium. A smoothing mechanism is introduced during the solution process to avoid drastic strategy oscillations, and an experience pool is utilized to store historical interaction data to accelerate learning. When sudden changes occur in the environment, agents can quickly adjust their own behavior based on online estimations of opponents' strategies, enabling the entire system to maintain robust collaborative performance under different operating conditions[4].

3.2 Local Strategy Generation Based on Environmental Perception and Maintenance of Global Objective Consistency

Decision-making in a multi-agent system at the edge essentially relies on each agent's observation of its local environment, while the global objectives (such as the real-time performance, integrity, or energy efficiency of data processing) are typically manifested as the overall outcome of all agents' behaviors. Local strategy generation requires mapping limited perceptual information into specific actions while implicitly considering the global objective. Deep neural networks possess the ability to extract state representations from high-dimensional observations, and their output layer is designed as a strategy probability distribution, enabling agents to make decisions within a continuous action space. To enhance the adaptability of local strategies to the environment, an attention mechanism is introduced to dynamically focus on key sensor data or the state information of neighboring agents, endowing the strategy generation process with interpretability and selectivity.

Deviations may occur between local decisions and global objectives, particularly when communication is limited or some agents fail. The consistency maintenance mechanism embeds the global objective into the optimization goal of local strategies during the strategy learning phase by designing auxiliary loss functions or consensus constraints. Specifically, in addition to maximizing their own cumulative rewards, each agent also needs to minimize the discrepancy between its own strategy and those of its neighboring agents, or ensure that the joint action satisfies certain global constraint conditions. For dynamically changing industrial processes, adopting the concept of model predictive control, agents locally optimize a sequence of decisions over a future rolling horizon and exchange prediction results with neighbors, correcting local deviations through negotiation. This distributed negotiation mechanism enables the entire system to effectively approach the global objective without a central node.

3.3 Effectiveness Evaluation and Adaptive Optimization of Collaborative Decision-Making Based on Policy Gradients

The effectiveness evaluation of collaborative decision-making needs to be measured from multiple dimensions, including indicators such as task completion quality, resource consumption levels, and system robustness. Policy gradient methods directly optimize policy parameters. By sampling interaction trajectories and calculating the gradient of cumulative rewards, they can handle problems with high-dimensional continuous action spaces. In multi-agent scenarios, the Centralized Training with Decentralized Execution framework allows for the use of global information during the training phase to evaluate the value of joint actions, thereby providing each agent with a more accurate policy gradient signal. In specific implementation, a centralized critic network is constructed, which takes as input the observations and actions of all agents and outputs an evaluation value for the current joint policy. Each agent's actor network then adjusts its own policy parameters based on this evaluation value. This architecture avoids the non-stationarity problem during training while maintaining complete distributivity during the execution phase.

The adaptive optimization mechanism enables the system to continuously adapt to changes in the industrial environment. When a decline in decision-making effectiveness or a drift in environmental statistical properties is detected, agents automatically adjust the learning rate, exploration noise, or network structure to re-converge to a better strategy. Meta-learning based methods allow agents to learn how to quickly adapt to new tasks from a series of similar tasks, significantly reducing the strategy adjustment time under new operating conditions. Furthermore, by introducing intrinsic motivation signals, agents are encouraged to explore parts of the state space that have not been sufficiently tried, preventing the strategy from falling into local optima. The effectiveness evaluation results are simultaneously fed back to the higher-level task scheduling module, providing a basis for task priority adjustment and resource reallocation, thus forming a closed-loop adaptive system from perception and decision-making to optimization.

Conclusion

This study has constructed a framework for multi-agent collaborative data processing and decision-making tailored for the edge environment of the Industrial Internet of Things. At the architectural level, the collaboration of heterogeneous nodes is achieved through multi-agent modeling, and distributed consensus mechanisms along with event-driven protocols are designed to address issues of data synchronization and interaction efficiency. At the model level, deep reinforcement learning is introduced to enable online feature extraction and dynamic task scheduling, while knowledge distillation and model quantization are integrated to overcome edge resource constraints. At the decision-making level, game theory and policy gradient methods are combined to establish a dynamic collaboration mechanism, balancing individual rationality with global objectives and enhancing system robustness. Future research will explore the integration of federated learning with multi-agent systems to enhance generalization capabilities under privacy preservation, investigate a cloud-edge-device collaborative optimization architecture, design scalable communication topologies for large-scale scenarios, and introduce explainable artificial intelligence to improve decision-making transparency.

Fund Projects

This paper is the "14th Five-Year Plan" project of the 2025 Collaborative Education Society: Research on the "Post-Course-Competition-Certificate-Innovation" Education Mechanism for Software Engineering Majors Driven by AI and Emerging Engineering Education (kt2025100118390856044)

References

[1] Tang Yue. "Design and Implementation of a Real-Time Data Processing System for the Industrial Internet of Things Based on Edge Computing." Yangtze River Information Communication, 38.07(2025): 225-227.
[2] Li Hao. Research on Joint Offloading and Caching Based on Multi-Agent and Dual Time Scales in the Industrial Internet of Things. 2025. Guilin University of Electronic Technology, MA thesis.
[3] Du Xiaojun. "Research on Optimization Strategies for Intelligent Control Systems Based on the Industrial Internet of Things." Software, 45.10(2024): 30-32.
[4] He Jingjian. Design and Implementation of an IoT Data Platform Based on Thing Models and Rule Engines. 2022. Automation Research and Design Institute of Metallurgical Industry, MA thesis.