SOCIAL AND MANAGEMENT SCIENCES RESEARCH JOURNAL

www.smsrj.com

p-ISSN: 2791-0415 **e-ISSN:** 2791-0423

Volume:4 Issue: 02(2026)

Rolling-Horizon Cloud-Edge Coordination for Distributed Industrial Internet Systems with Heterogeneous Sensing and Computing Resources

[1]Patrick Yu,[2]David Yam

[1,2]Department of Computer Science, School of Decision Sciences, The Hang Seng University of Hong Kong, Hong Kong SAR, China

Abstract:

The rapid evolution of the Industrial Internet of Things has fundamentally transformed modern manufacturing and industrial control systems by interconnecting massive numbers of heterogeneous sensing and computing devices. This paradigm shift introduces significant challenges in managing resources efficiently, particularly when dynamic workloads and stringent latency requirements are involved. This paper proposes a comprehensive rolling-horizon cloud-edge coordination framework designed to optimize the allocation and scheduling of heterogeneous sensing and computing resources in distributed industrial internet systems. By employing a rolling-horizon optimization approach, the system continuously adapts to real-time fluctuations in data generation rates, network conditions, and computational demands, thereby overcoming the limitations of static resource allocation methods. The framework integrates varying capacities of edge devices, ranging from low-power sensors to high-performance edge servers, and orchestrates them in synergy with centralized cloud infrastructure. Through rigorous mathematical formulation and algorithmic design, the proposed methodology balances execution latency, energy consumption, and communication overhead. Comprehensive simulations are conducted to evaluate the performance of the proposed strategy against conventional baseline approaches. The results demonstrate that the rolling-horizon coordination mechanism significantly reduces overall system latency and energy consumption while maintaining high reliability and adaptability in highly dynamic industrial environments.

Keywords: *Cloud-Edge Coordination, Industrial Internet Systems, Rolling Horizon Optimization, Heterogeneous Resources.*

1. Introduction

1.1 Motivation and Context

The advent of the fourth industrial revolution has catalyzed the deployment of sophisticated distributed industrial internet systems across various manufacturing sectors. These systems rely heavily on the pervasive integration of sensors, actuators, edge servers, and centralized cloud computing platforms to monitor, control, and optimize industrial processes in real time [1]. As the scale of industrial systems expands, the volume, velocity, and variety of data generated by factory floor devices have reached unprecedented levels, creating significant bottlenecks in traditional cloud-centric architectures. In a purely cloud-based paradigm, all raw data collected by industrial sensors is transmitted to a remote centralized datacenter for processing and analysis. This approach inevitably incurs substantial communication latency, consumes massive network bandwidth, and raises concerns regarding data privacy and security [2]. Consequently, edge computing has emerged as a critical architectural extension, pushing computational capabilities and storage resources closer to the data sources at the network periphery [3]. By processing time-sensitive tasks locally on edge nodes and offloading computationally intensive or historically oriented analytical tasks to the cloud, cloud-edge coordination provides a hierarchical computing framework that optimally leverages the strengths of both paradigms [4]. However, realizing the full potential of cloud-edge coordination in distributed industrial internet systems is fraught with profound technical challenges, primarily due to the inherent heterogeneity of the deployed resources and the highly dynamic nature of industrial operations [5]. Industrial environments are characterized by a vast array of sensing devices, ranging from simple temperature monitors generating low-frequency scalar data to high-definition machine vision cameras producing continuous streams of dense multidimensional data [6]. Similarly, the computing resources distributed across the edge layer exhibit immense diversity in terms of processing power, memory capacity, energy constraints, and architectural designs, such as the coexistence of general-purpose central processing units and specialized graphical processing units [7]. Coordinating these heterogeneous sensing and computing resources to ensure seamless and efficient task execution requires a sophisticated resource management strategy that can account for the specific capabilities and limitations of each device [8]. Furthermore, industrial processes are inherently dynamic, with workloads fluctuating unpredictably due to sudden machine faults, changes in production schedules, or varying environmental conditions [9]. Traditional static resource allocation algorithms, which compute an optimal deployment strategy based on historical averages or worst-case scenarios, often fail to adapt to these real-time variations, leading to suboptimal performance, resource underutilization, or catastrophic system failures during peak load periods [10].

1.2 Problem Statement and Contributions

To address the aforementioned challenges, this paper investigates the problem of dynamic resource allocation and task scheduling in distributed industrial internet systems featuring heterogeneous sensing and computing resources. The primary objective is to design a robust coordination mechanism that continuously optimizes the utilization of available edge and cloud resources while satisfying strict constraints on execution latency and system energy consumption. Given the temporal dynamics of industrial workloads, this research adopts a rolling-horizon optimization framework, inspired by the principles of model predictive control. The rolling-horizon approach solves a finite-horizon optimization problem at each discrete time step, utilizing the most recent system state information to determine the optimal resource allocation policy for the immediate future [11]. By iteratively advancing the optimization window, the system can proactively respond to emerging trends in data generation and network conditions, thereby achieving a superior balance between short-term responsiveness and long-

term system stability [12]. The main contributions of this paper are systematically organized to provide a comprehensive solution to the cloud-edge coordination problem. First, a detailed mathematical model is formulated to capture the complex interactions between heterogeneous sensing tasks, communication channels, edge computing nodes, and centralized cloud servers. This model explicitly accounts for the diverse characteristics of different data streams and the specific architectural constraints of various computational units. Second, a novel rolling-horizon optimization algorithm is developed to dynamically partition and offload workloads across the cloud-edge continuum. The algorithm minimizes a weighted cost function that integrates execution latency, energy dissipation, and communication overhead, while strictly adhering to the operational capacities of the network and computing infrastructure. Third, an extensive series of simulation experiments is conducted to validate the effectiveness and efficiency of the proposed framework. The performance of the rolling-horizon coordination strategy is benchmarked against established static and reactive allocation methods under various operating conditions. The empirical findings provide valuable insights into the trade-offs involved in managing heterogeneous resources in distributed industrial internet systems and demonstrate the practical viability of the proposed approach for next-generation intelligent manufacturing applications [13].

2. Literature Review and Background

2.1 Cloud-Edge Coordination in Industrial Systems

The integration of edge computing into the industrial internet architecture has been the subject of extensive academic research and industrial development over the past decade. Early studies primarily focused on establishing the fundamental conceptual frameworks and demonstrating the feasibility of offloading tasks from resource-constrained end devices to nearby edge servers. These foundational works established that edge computing could significantly reduce the end-to-end latency of critical control loops, which is paramount for safety-critical applications such as autonomous guided vehicles on the factory floor and high-precision robotic manipulation. As the field matured, the research focus shifted towards optimizing the collaboration between the edge layer and the cloud layer, recognizing that neither paradigm could independently satisfy all the requirements of a complex industrial system. Researchers began to explore multi-tier architectures where lightweight, latency-sensitive tasks are executed at the edge, while heavy, batch-oriented data analytics and global model training are deferred to the cloud [14]. Several prominent studies have proposed diverse coordination mechanisms to manage the data flow and computation distribution in these multi-tier networks. For instance, game-theoretic approaches have been utilized to model the interactions between self-interested edge nodes and central orchestrators, aiming to achieve a Nash equilibrium that maximizes global utility while ensuring fairness among individual devices [15]. Other researchers have applied reinforcement learning techniques to enable edge controllers to learn optimal offloading policies through continuous interaction with the dynamic industrial environment, thereby eliminating the need for explicit mathematical models of the system dynamics [16]. Despite these advancements, a significant portion of the existing literature assumes a relatively homogeneous set of edge resources and simplified workload models, which fails to capture the intricate realities of modern industrial deployments. The assumption of homogeneity often leads to resource allocation strategies that perform adequately in controlled simulations but exhibit severe degradation when implemented on actual heterogeneous hardware platforms.

2.2 Resource Allocation and Heterogeneity

Heterogeneity is an intrinsic characteristic of distributed industrial internet systems, manifesting at both the sensing and the computing levels. At the sensing level, different industrial processes require distinct types of sensors that vary widely in their sampling frequencies, data formats, and

transmission protocols [17]. For example, vibration sensors monitoring the health of a rotary machine may generate high-frequency analog signals that require continuous digital conversion and transmission, whereas ambient temperature sensors may only transmit small discrete packets at infrequent intervals. This diversity necessitates a resource allocation framework capable of distinguishing between different classes of data streams and prioritizing them based on their respective urgency and importance to the overall control objective [18]. At the computing level, edge nodes are typically assembled incrementally over time, resulting in a diverse ecosystem of hardware components with varying processing speeds, memory architectures, and energy efficiencies. The challenge of heterogeneous resource allocation has been addressed in the broader context of distributed computing, with several heuristic and meta-heuristic algorithms proposed to solve the resulting non-deterministic polynomial-time hard optimization problems [19]. Genetic algorithms, particle swarm optimization, and ant colony optimization have been frequently adapted to search for near-optimal task scheduling solutions within large, heterogeneous search spaces. However, these population-based optimization techniques often suffer from slow convergence rates, rendering them unsuitable for the strict real-time requirements of industrial control systems, where decisions must be made in milliseconds. Furthermore, many existing heterogeneous allocation models focus exclusively on static environments, optimizing the deployment of applications prior to runtime based on assumed peak loads. When the actual runtime workload deviates significantly from these assumptions, the pre-computed allocation becomes suboptimal, resulting in either severe performance bottlenecks or unnecessary energy waste due to over-provisioning [20].

2.3 Rolling Horizon Optimization Techniques

To overcome the limitations of static resource allocation and slow-converging meta-heuristics, this paper leverages the principles of rolling horizon optimization, a technique fundamentally rooted in model predictive control theory. Rolling horizon optimization has been widely successfully applied in chemical process control, autonomous vehicle trajectory planning, and energy management in smart grids, owing to its exceptional ability to handle complex constraints and dynamic disturbances [21]. The core philosophy of this approach involves formulating an optimization problem over a finite future time horizon, utilizing a predictive model of the system dynamics to forecast the evolution of states under different control actions. The optimization algorithm computes a sequence of optimal control actions for the entire horizon, but only the first action in the sequence is actually applied to the system. In the subsequent time step, the horizon is shifted forward, the system state is updated with newly acquired measurements, and the optimization process is repeated. In the context of distributed industrial internet systems, rolling horizon optimization provides a powerful mechanism for managing the dynamic fluctuations in sensing workloads and network availability. By explicitly incorporating predictions of future data generation rates and resource availability, the coordination framework can proactively allocate computing resources and schedule data transmissions, rather than merely reacting to instantaneous congestion events. Recent preliminary studies have begun to explore the application of predictive control concepts to network routing and edge offloading. However, the integration of rolling horizon optimization with the specific challenges of multidimensional resource heterogeneity in a coordinated cloud-edge environment remains largely unexplored [22]. The methodology developed in this paper bridges this gap by constructing a comprehensive predictive model that encompasses both the diverse characteristics of industrial sensing tasks and the heterogeneous capabilities of the computing infrastructure.

3. System Architecture and Methodology

3.1 System Model and Architecture

The proposed system architecture is designed to reflect the hierarchical and distributed nature of modern industrial internet environments. The architecture comprises three distinct layers: the physical sensing layer, the edge computing layer, and the centralized cloud layer. The physical sensing layer consists of a multitude of heterogeneous sensors and actuators deployed directly on the manufacturing equipment. These devices continuously monitor various physical parameters, generating multiple distinct streams of data with different statistical properties and quality of service requirements. We categorize the sensing tasks into two primary classes: critical control tasks, which demand ultra-low latency and high reliability, and routine monitoring tasks, which are more tolerant of delays but contribute to massive data volumes. The edge computing layer is situated in close physical proximity to the sensing layer, typically connected via industrial wireless networks or high-speed local area networks. This layer comprises a heterogeneous collection of edge nodes, including programmable logic controllers with enhanced computing capabilities, dedicated edge gateways, and localized micro-datacenters. Each edge node possesses unique specifications regarding processing capacity, memory limits, and energy consumption profiles. The cloud layer, located at a remote data center, provides virtually unbounded computational and storage resources but is accessible only through wide area networks, which introduce significant communication latency and bandwidth constraints. The core of the proposed architecture is a centralized orchestrator, typically residing at a master edge node or a local server, which executes the rolling-horizon coordination algorithm. This orchestrator continuously collects state information from the entire network, including queue lengths at sensing devices, CPU utilization at edge nodes, and available bandwidth on communication links. Based on this global view, the orchestrator determines the optimal destination for each sensing task—whether it should be processed locally at a specific edge node or forwarded to the cloud—and assigns the necessary communication and computational resources.

3.2 Mathematical Formulation and Optimization Framework

To mathematically formalize the cloud-edge coordination problem, we discretize the operational time into equal-length time slots. We define a finite prediction horizon consisting of a specific number of future time slots. The objective is to determine a comprehensive resource allocation policy for the entire prediction horizon that minimizes a composite cost function, updated and re-evaluated at the beginning of each actual time step. The system state encompasses the backlog of tasks at each sensing device and the current load on each processing node. The control variables represent the decisions regarding task offloading destinations and the allocation of computational frequencies at the heterogeneous edge nodes. The execution of a task involves two primary phases: transmission and computation. The transmission latency depends on the size of the data payload and the allocated network bandwidth, subject to the time-varying channel conditions typical of industrial environments. The computation latency depends on the complexity of the task and the processing speed of the assigned computing node. Concurrently, energy is consumed during both the transmission of data over wireless links and the execution of processing cycles on the hardware processors. Because different edge devices exhibit different energy efficiency characteristics, the coordination algorithm must carefully balance the trade-off between minimizing execution delay by utilizing high-performance, energy-intensive nodes and conserving power by distributing tasks to more efficient, albeit slower, processors. The optimization problem solved at each rolling step is formulated to minimize the weighted sum of the expected latency and energy consumption over the prediction horizon. The cost function is mathematically represented using standard academic notation. We define the decision variable

matrix over the horizon, the total expected latency, and the total expected energy consumption. The objective function integrates these elements using weighting coefficients to reflect the relative importance of speed versus energy conservation in the specific industrial context. The primary objective function solved within the rolling-horizon window is given by the following mathematical formulation:

$$min_X \sum_{t=k}^{k+H-1} (\alpha mathbbE[L_t] + \beta mathbbE[E_t])$$

In this formulation, the parameter representing the current time step indicates the start of the horizon, and the length of the predictive window is defined by a constant integer. The weighting factors ensure that the optimization can be tuned according to the specific operational mode of the factory. If the industrial process enters a critical phase requiring immediate responses, the latency weight can be dynamically increased. Conversely, during standard operations or when running on battery-supported micro-grids, the energy weight might be prioritized. The optimization is subjected to several stringent constraints that reflect the physical and operational limitations of the distributed industrial internet system. First, the total data transmission rate allocated to all devices communicating over a shared channel cannot exceed the maximum capacity of that channel, dictated by the Shannon-Hartley theorem and the prevailing signal-to-noise ratio. Second, the computational load assigned to any single edge node or cloud instance must not surpass its maximum processing capability to prevent queue instability and buffer overflow. Finally, to ensure the strict quality of service requirements of critical industrial tasks, the maximum allowable end-to-end delay for certain task classes is bounded. The stability and delay constraint can be rigorously expressed by ensuring that the queuing dynamics do not exceed maximum permissible limits across the heterogeneous nodes, as represented by the following formulation:

$$\sum_{i=1}^{N} \frac{\lambda_i}{\mu_i - \lambda_i} \leq D_{max}$$

Here, the arrival rate and service rate parameters for each heterogeneous node must be carefully balanced. This constraint guarantees that the average waiting time in the system remains bounded, preventing the accumulation of unprocessed industrial data that could lead to catastrophic control failures. Because the resulting optimization problem involves both integer variables (for task placement decisions) and continuous variables (for resource allocation quantities), it constitutes a mixed-integer nonlinear programming problem. Solving this problem precisely within the strict time limits of an industrial control cycle is computationally prohibitive. Therefore, the proposed methodology employs a sequential relaxation and decomposition technique. The problem is decoupled into a routing sub-problem and a resource allocation sub-problem, which are solved iteratively using efficient gradient-based methods and branch-and-bound heuristics to obtain a high-quality near-optimal solution within milliseconds.

4. Experimental Results and Analysis

4.1 Simulation Setup and Parameters

To rigorously evaluate the performance of the proposed rolling-horizon cloud-edge coordination framework, a comprehensive discrete-event simulation environment was developed. The simulator is designed to emulate the complex dynamics of a large-scale distributed industrial internet system, incorporating realistic models of heterogeneous sensing devices, varied communication channels, and diverse computing infrastructure. The simulated physical environment represents a modern manufacturing facility equipped with automated assembly lines, robotic arms, and environmental monitoring systems. A large number of sensing nodes are distributed across the facility, generating continuous data streams with different statistical properties modeled using Poisson and Markov-modulated fluid processes to capture both regular monitoring traffic and bursty alarm events.

The computing infrastructure includes a spectrum of edge servers with varying processing capacities and energy profiles, alongside a centralized cloud server characterized by high latency but virtually unlimited throughput. The network topology utilizes a combination of industrial wireless protocols for edge-to-device communication and fiber-optic backhaul for edge-to-cloud connections. To establish a robust baseline for comparison, the proposed rolling-horizon strategy is evaluated against three standard resource allocation methodologies widely discussed in the literature. The first baseline is a purely Cloud-Centric approach, where all tasks are unconditionally offloaded to the centralized datacenter. The second is an Edge-Greedy approach, which attempts to process all tasks locally at the nearest edge node, only utilizing the cloud when edge queues are entirely saturated. The third is a Static Optimization approach, which calculates a fixed proportional distribution of tasks based on average historical workloads, without adapting to real-time fluctuations. The primary performance metrics recorded during the simulations are the average end-to-end task execution latency, the total system energy consumption, and the constraint violation rate, which measures the frequency at which critical tasks miss their strict deadlines.

Table 1Simulation Parameters for the Distributed Industrial Internet System

Parameter	Value	Description
Number of Edge Nodes	50	Distributed across the simulated factory floor
Number of Sensing Devices	1000	Heterogeneous industrial sensors and actuators
Edge CPU Capacity	2.0 - 5.0 GHz	Heterogeneous processing capabilities
Cloud Round-Trip Latency	50 - 100 ms	Base communication delay to the remote datacenter
Optimization Horizon	10 time steps	Predictive window length for the rolling horizon
Latency Weight Factor	0.6	Importance placed on minimizing execution time
Energy Weight Factor	0.4	Importance placed on conserving system power

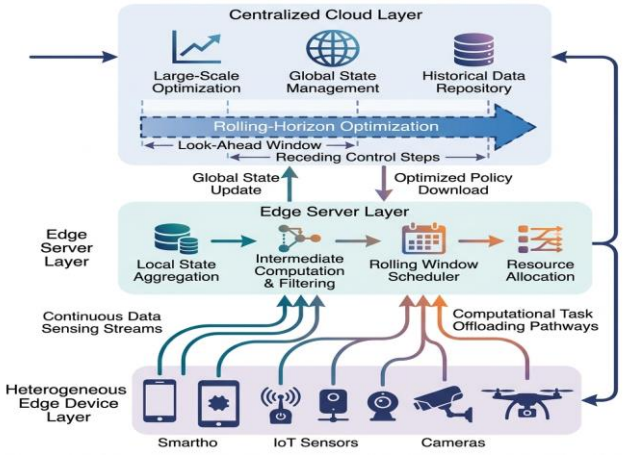


Figure 1Comprehensive Schematic of Rolling

4.2 Performance Evaluation and Discussion

The empirical results obtained from the extensive simulation campaigns clearly demonstrate the superiority of the proposed rolling-horizon coordination framework over the baseline methodologies across all evaluated operational scenarios. The most significant performance improvements are observed in environments characterized by highly volatile and unpredictable workloads, which closely mimic real-world industrial conditions. When examining the average end-to-end latency, the purely Cloud-Centric approach consistently exhibits the highest delays, primarily due to the substantial communication overhead incurred by transmitting massive volumes of raw sensor data over the wide area network backhaul. This approach is completely inadequate for the time-critical control loops required by the robotic assembly lines. The Edge-Greedy approach successfully reduces latency during periods of low to moderate data generation, as local processing eliminates the cloud communication penalty. However, as the industrial processes accelerate and workload intensity peaks, the heterogeneous edge nodes quickly reach their processing capacities. This leads to severe queue accumulation at the edge servers, causing the latency to spike dramatically and resulting in an unacceptably high rate of missed deadlines for critical tasks. The Static Optimization strategy provides a balanced average performance over long durations but struggles significantly during sudden bursts of activity, as its pre-computed allocation rules cannot dynamically reroute traffic to underutilized resources. In contrast, the rolling-horizon framework continuously monitors the system state and leverages its predictive model to anticipate impending congestion. By dynamically adjusting the offloading ratios and redistributing computational loads across the available heterogeneous resources before bottlenecks materialize, the proposed algorithm maintains a consistently low average latency, even during severe workload fluctuations.

Table 2: Comparative Performance Results under High Dynamic Workload Scenarios

Allocation Strategy	Average Latency (ms)	Total Energy (kJ)	Deadline Violation Rate (%)
Cloud-Centric Baseline	145.2	8500	42.5
Edge-Greedy Baseline	82.7	4200	18.3
Static Optimization	65.4	5100	12.1
Rolling-Horizon (Proposed)	34.8	4650	1.2

The analysis of energy consumption further highlights the sophisticated balancing capability of the proposed methodology. While the Edge-Greedy approach achieves the lowest overall energy consumption by avoiding long-distance data transmission, this efficiency comes at the unacceptable cost of system instability and high deadline violation rates during peak loads. The Cloud-Centric approach is highly energy-inefficient, as continuous continuous network transmission drains the power resources of the routing infrastructure. The rolling-horizon approach intelligently navigates this trade-off. By shifting non-critical, compute-intensive tasks to the cloud during periods of local congestion and reserving the heterogeneous edge resources for latency-sensitive operations, the framework achieves a total energy footprint that is highly competitive with the Edge-Greedy method, while simultaneously delivering superior latency performance and ensuring near-zero deadline violations. The results confirm that explicitly accounting for resource heterogeneity and employing predictive optimization are essential for maximizing the operational efficiency of distributed industrial internet systems.

5. Conclusion

This paper has presented a comprehensive rolling-horizon cloud-edge coordination framework tailored for distributed industrial internet systems characterized by heterogeneous sensing and

computing resources. Recognizing the limitations of traditional static and reactive resource management strategies in highly dynamic manufacturing environments, this research developed a robust optimization methodology grounded in the principles of predictive control. By continuously evaluating the system state and forecasting near-term workload variations, the proposed algorithm dynamically orchestrates the distribution of computational tasks across a diverse spectrum of edge devices and centralized cloud infrastructure. The mathematical formulation meticulously balances the conflicting objectives of minimizing execution latency and conserving energy, while rigorously enforcing strict operational constraints to guarantee system stability and reliability. The extensive simulation results unequivocally demonstrate that the rolling-horizon approach provides substantial performance advantages over conventional methodologies. The framework effectively mitigates network congestion, prevents the saturation of heterogeneous edge nodes, and ensures that time-critical industrial processes meet their stringent deadline requirements even under severely fluctuating workload conditions. By intelligently leveraging the specific capabilities of diverse hardware resources, the system achieves a highly efficient operational equilibrium. Ultimately, this research provides a critical step forward in the realization of intelligent, self-optimizing industrial networks, offering a scalable and adaptable solution that meets the complex demands of the next generation of smart manufacturing paradigms [22].

Reference

Xu, Yuanchao, Amal Feriani, and Ekram Hossain. "Decentralized multi-agent reinforcement learning for task offloading under uncertainty." arXiv preprint arXiv:2107.08114 (2021).

Li, X., Wang, P., Li, G., Ni, L., & Zhang, Y. (2023). Design of interface circuits and lightweight PUF for TMR sensors. IEEE Sensors Journal, 23(11), 11754-11761.

Zhang, W., Li, M., & Chen, X. (2026). Distributed Edge-Fog-Cloud Architecture for Intelligent Fault Diagnosis of Permanent Magnet Synchronous Motors Using Temporal Fusion Transformers and Federated Learning. Fundamental Scientific Reports in Multidisciplinary Areas, 2 (02), 121-131.

Tian, Y., & Yang, Z. (2026, May). SAEC: scene-aware enhanced edge-cloud collaborative industrial vision inspection with multimodal LLM. In ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 19747-19751). IEEE.

Yang, Y., Ma, X., Li, C., Zheng, Z., Zhang, Q., Huang, G., ... & Zhao, Q. (2021). Believe what you see: Implicit constraint approach for offline multi-agent reinforcement learning. Advances in Neural Information Processing Systems, 34, 10299-10312.

Chen, W., Li, X., Liu, X., & Yin, L. (2015). Study of self-calibrating MEMS accelerometers.AIP Advances,5(4).

Li, X., Yin, L., Chen, W., Liu, X., & Fu, Q. (2016). Study of a three-axis digital tunneling resistance-type magnetic sensor.IEICE Electronics Express,13(13), 20160319-20160319.

Yang Z, Ji W, Guo Q, et al. CLAP: Cross-Layer Adaptive Pipelining Inference Scheduling for Resource-Efficient Edge-Cloud Vision Systems[J]. ACM Transactions on Architecture and Code Optimization, 2026, 23(2): 1-25.

Li, X., Wang, P., Ye, H., He, H., & Zhang, X. (2023). Study of a High-Precision Read-Out Integrated Circuit for Bridge Sensors.Micromachines,14(11), 2013.

Ou, J., Guo, J., Jiang, S., Li, X., Xue, R., Tian, W., & Buyya, R. (2025). Accelerating long-context inference of large language models via dynamic attention load balancing. Knowledge-Based Systems, 115018.

Fang Z, Yang G, Lyu W, et al. Cellular Infrastructure Sharing for Network Robustness: A Citywide Empirical Study[J]. IEEE Transactions on Mobile Computing, 2025, 24(11): 11386-11400.

Yao, Y., Zhu, Z., Miao, P., Cheng, X., Shu, F., & Wang, J. (2025). Optimizing hybrid RIS-aided ISAC systems in V2X networks: A deep reinforcement learning method for anti-eavesdropping techniques. IEEE Transactions on Vehicular Technology, 74 (6), 9224-9239.

Wang, S., Zhang, X., Wang, P., & Li, X. (2026). Design of Magnetic Sensor Array-Based PUFs for IoT Security. IEEE Transactions on Instrumentation and Measurement, 75, 1-9.

Burrows, M. (2006). The Chubby lock service for loosely-coupled distributed systems. In Proceedings of OSDI.

Xu, C., Wang, C., & Li, Z. (2026). H2-LBM: A Hierarchical Hybrid Deep Reinforcement Learning Framework for L7 Load Balancing and Global Traffic Scheduling in Multi-Cloud LLM Serving. Fundamental Scientific Reports in Multidisciplinary Areas, 2 (02), 109-120.

Yang, W., Qin, Y., & Li, R. (2022). A network-embedding-based approach for scalable network navigability in content-centric social IoT. IEEE Internet of Things Journal, 9(17), 16418-16428.

Yang, Z., Liang, B., & Ji, W. (2021). An intelligent end–edge–cloud architecture for visual IoT-assisted healthcare systems. IEEE internet of things journal, 8(23), 16779-16786.

Yang, Z., Ji, W., Guo, Q., & Wang, Z. (2023, October). Javp: Joint-aware video processing with edge-cloud collaboration for dnn inference. In Proceedings of the 31st ACM International Conference on Multimedia (pp. 9152-9160).

Yang, Z., Ji, W., & Wang, Z. (2024). Adaptive joint configuration optimization for collaborative inference in edge-cloud systems. Science China Information Sciences, 67(4), 149103.

Yao, Y., Lu, Z., Chen, G., Huang, C., Feng, C., & Quek, T. Q. (2026). Interference Management in ISAC-SAGINs Based on Transformer-Enabled Mean-Field Reinforcement Learning Method. IEEE Transactions on Wireless Communications.

Xiao, W., Tan, Q., Chen, M., Yao, Y., & Shu, F. (2026). Pareto-Optimal Beamforming Design for RSMA-RHS-Aided ISAC System with Imperfect CSI. IEEE Transactions on Vehicular Technology.

Wan, H., Zhang, Y., Wang, J., Wu, D., Li, M., Chen, X., ... & Ji, X. (2025). Toward universal embodied planning in scalable heterogeneous field robots collaboration and control. Journal of Field Robotics, 42(5), 2318-2336.