

Research on Design and Performance Optimization of RDMA-based Zero-Copy Message Middleware for High-Frequency Trading Matching Links

Wenjie Jiang

Guangzhou College of Commerce, School of Information Technology & Engineering, Guangzhou, 511363

Keywords: High-frequency trading; RDMA; Zero-copy; Message broker; Low-latency network

Abstract: High-frequency trading systems impose requirements on message middleware, including sub-microsecond software overhead, strict sequential semantics, extremely low tail latency, and predictable jitter control. Traditional middleware based on kernel sockets or general message queues typically incurs overhead such as user-space-kernel copying, protocol stack traversal, and interrupt/context switching, making it difficult to maintain stable latency under extreme market fluctuations. To address this issue, this paper proposes a zero-copy message middleware architecture based on RDMA for high-frequency trading scenarios: On the data plane, a pre-registered memory pool, a single-producer, single-consumer ring descriptor, two-level batch processing, and lock-free completion queues are employed; on the control plane, lightweight connection management, credit backhaul, priority channels, and fast failover mechanisms are introduced; and on the message semantic layer, bounded sequential submission, retransmission window constraints, and bypass replication of matching events are implemented. Combining RDMA research findings from the past three years with publicly available experimental data, a latency model, a zero-copy benefit model, and a service capability constraint model are constructed, and performance analysis for multiple scenarios is presented. Research shows that the coordinated optimization of zero-copy and kernel bypass can significantly reduce CPU usage and tail latency; through the coordinated tuning of message granularity, queue depth, and batch processing threshold, throughput stability can be improved while maintaining sequential semantics. This paper's design provides an engineeringable middleware implementation approach for high-frequency trading message buses, market data distribution, and risk control linkage.

1 Introduction

In high-frequency trading environments, the transmission of market data, order placement, risk control review, and reward delivery before and after matching all rely on low-latency messaging systems to achieve end-to-end collaboration. Because market price discovery is highly spontaneous,

correlated, and time-sensitive, message brokers must not only minimize average latency but also keep the 99th and 99.9th percentile latency controllable. Especially in scenarios like pre-market auctions, macroeconomic data releases, and cross-market linkages, any additional copying, kernel scheduling delays, or queue congestion in the message path will lengthen order queuing times and reduce strategy profitability.

In the past three years, research on RDMA high-performance communication, software programmable control, connection management and other related technologies has been deepening, and the academic and industrial communities have been conducting related research. Flor believes that on the basis of retaining the hardware of the RDMA data plane, a part of the control plane should be put into software to adapt to heterogeneous RNICs and lossy Ethernet networks[1]; sRDMA analyzed the insufficient RDMA resources under the condition of massive connections in terms of scheduling[2]; Harmonic and PeRF started from the two aspects of isolation and preemption, and tried to improve the availability and stability of RDMA in public cloud and multi-tenant environments[3-4]. Zero-sided RDMA, SwCC, White-Boxing RDMA and UCM and other works have shown from a higher level that RDMA is no longer a low-level remote access, but a new infrastructure that supports message passing, software control, connection orchestration and maintainability design[5,7-9].

However, most existing research focuses on general cloud computing, databases, or training clusters, rarely addressing the mid-level system design from a middleware perspective to address the combined constraints of "extremely short messages + high-frequency bursts + strict ordering + fault-tolerant bypassing" in high-frequency trading. Messages in high-frequency trading typically exhibit characteristics such as small message bodies, long synchronization links, tight timeframes, and high packet loss costs. Directly using a general RDMA framework can easily generate new tail-latency hotspots in connection establishment, memory registration, queue polling, and message processing control. Therefore, it is necessary to re-examine the high-frequency trading message chain, re-evaluate the architectural boundaries of RDMA zero-copy middleware, and develop a consistency ordering guarantee mechanism that conforms to transaction semantics.

Against this backdrop, this paper proposes a design scheme for a zero-copy message middleware based on RDMA, specifically for high-frequency trading, and demonstrates its effectiveness from three aspects: architecture, algorithm, and performance model. The research objectives are to establish a low-jitter, scalable data path, reduce the cost of user-space message assembly and copying, ensure orderly data transmission according to the needs of matching services, provide adaptable sequencing, retry, and bypass replication mechanisms, and offer methods for parameter tuning and suggestions for engineering implementation. The paper is organized in the following order: introduction, analysis of the current research status, problem statement, problem-solving or strategy, and conclusion.

2. Current Status Analysis of the Research Topic

The development of RDMA technology and the reshaping of middleware value. RDMA puts the functions involved in data transmission, such as message segmentation, DMA read and write, and reliable transmission, into RNIC, allowing applications to bypass the kernel protocol stack and directly operate on remote memory, which greatly reduces software overhead. Compared with traditional TCP sockets, RDMA has advantages such as zero copy, kernel bypass, low CPU usage, and more stable round-trip latency, so it is widely used in distributed storage, memory databases, training clusters, and RPC in data centers [1][6]. The buffering, forwarding and copying work that was originally done by the broker and kernel network stack in the message middleware can now be implemented in a lightweight way using pre-registered memory, descriptors and remote notification, which greatly reduces the path that messages take.

Transform the general RDMA framework into software programmable control. Due to the increase in network speed and the complexity of load patterns, congestion control and connection management that are fixed in the hardware can no longer adapt to the development and changes of services. SwCC shows that after adding a programmable control core on the NIC side, software-defined congestion control can be achieved in about 3.1 microsecond control loop time [7]; White-Boxing RDMA further proposes to perform packet-granular software control on hardware transmission, so that the software can obtain more fine-grained observability and schedulability [8]. The progress of high-frequency trading is that message middleware no longer relies entirely on the behavior of black-box RNIC, and adopts strategies that are more like transaction traffic in places such as credit feedback, flow control threshold, and abnormal backoff.

Issues related to quality, reliability and large-scale deployment. Although high-frequency trading systems are generally deployed on controllable private networks or low-latency networks in the same data center, the establishment of connections and reliable transmission remain major issues in situations such as disaster recovery, cross-engine synchronization, and multi-node risk control linkage. UCM research points out that the maintainability of user-space RDMA connection establishment mechanisms is just as important as connection latency [9]; Mitigating Scalability Walls of RDMA-based Container Networks reveals the implicit bottlenecks brought about by QP, CQ and address mapping under scale expansion [10]. In addition, SDR-RDMA and others have shown that in situations with greater distances or more complex faults, traditional selective retransmission strategies are no longer sufficient and more flexible software-defined reliability expansion is required.

Existing solutions lack adaptability for high-frequency trading. Most research focuses on large message transmission or multi-tenant fairness in cloud environments. For high-frequency trading, the emphasis is on nanosecond or microsecond-level processing methods for short messages ranging from a few hundred to a few thousand characters. Secondly, general-purpose systems typically prioritize throughput, but high-frequency trading is also constrained by queue jitter, sequential submission, and abnormal bypass replication. Thirdly, existing literature on the complex chain of "market data → strategy → order placement → risk control → reward" lacks a unified abstract form and cannot directly answer how message middleware collaborates with the trading matching engine under strict business timing. Therefore, there is still considerable room for system design in the research of RDMA zero-copy middleware for high-frequency trading.

3. Raise questions

Based on the preceding research and business needs, the problems existing in high-frequency trading scenarios can be summarized into four layers.

One issue is the excessively long data path. If a message enters the kernel from the application buffer, is copied to the NIC via the network stack, and then sent to the host by the NIC, this results in multiple memory copies of the message and multiple cache loss events. In short-message-dominated order and reward streams, the copying cost is not reduced by larger messages; instead, it becomes a major component of the overall latency.

Second, latency stability is poor. High-frequency trading focuses on tail distribution, not simply calculating the average. Even if the average latency is already very low, the lack of effective control over memory registration, completion notification, queue congestion, batch threshold selection, and retry logic can still cause significant fluctuations in p99 latency, thus affecting the preemptiveness of the strategy.

Third, sequential semantics and reliability cannot be satisfied simultaneously. There are causal relationships between messages in a trading system. Messages such as market snapshots, incremental updates, order confirmations, and transaction reports must be sent in a certain order.

However, traditional high-throughput middleware generally uses asynchronous batching, unordered write-back, or eventual consistency to improve throughput, which contradicts the semantics of trading.

Fourthly, the engineering maintenance is quite complex. RDMA development involves dealing with low-level details such as QP, CQ, MR, doorbell, and completion polling. If the middleware lacks a clear abstraction layer, then system iteration will inevitably rely on low-level network experts, which is not conducive to the reuse of trading strategies, risk control modules, and market data services.

Therefore, the core issue of this paper is how to design a high-performance middleware that can fully utilize the advantages of RDMA zero-copy and kernel bypass to meet the requirements of high-frequency transaction short messages, strict ordering, low jitter, and easy maintenance, and to provide specific performance optimization methods.

4. Problem Solving/Strategies

Overall Architecture Design. The middleware proposed in this paper is built using a three-layer structure: lightweight control plane, extremely simplified data plane, and transactional semantic layer. The data plane uses a pre-registered memory pool and a circular descriptor array to organize messages. The sender only writes the descriptor and payload address; the RNIC uses DMA to directly complete remote writes; and the receiver uses polling to perform uninterrupted packet reception by checking queues and local sequence numbers. The control plane implements connection establishment, buffer leases, credit feedback, heartbeats, and failover. The semantic layer provides abstractions such as Topic, Channel, Ordered Stream, Fast Ack, and Replay Cursor, allowing transaction services to use the middleware without exposing the details of the underlying verbs.

$$T_{eze} = L_{post} + L_{dma} + L_{nic} + L_{net} + L_{poll} \quad (1)$$

Equation (1) specifies the components that make up the end-to-end latency of a single message. This represents the overhead caused by the application request submission, the local DMA transfer time, the time taken for the RNIC to process the message, the delay in propagation to network nodes and exchange, and the time required for the receiving end to complete polling and acknowledgment. The optimization focus of high-frequency trading is to simultaneously reduce the values of each mean and their variance.

$$G(s) = s \cdot \alpha_{copy} - (\beta_{reg} + \beta_{desc} + \beta_{poll}) \quad (2)$$

Equation (2) is the benefit model of zero-copy compared to the traditional copy path. The values are: message length s , unit byte copy cost α_{copy} , memory registration amortization cost β_{reg} , descriptor maintenance overhead β_{desc} , and increased control overhead due to polling β_{poll} . When $G(s) > 0$, the zero-copy path is generally better. For short transaction messages, memory pool reuse and pre-registration should be used to reduce β_{reg} .

$$\lambda \leq \min\left(\frac{B_{nic}}{\bar{s}}, \frac{Q_d}{T_{srv}}\right) \quad (3)$$

Equation (3) provides the constraint on message arrival rate during stable system operation. Here, B_{nic} is the effective bandwidth of the network interface card, \bar{s} is the average message size, Q_d is the available queue depth, and T_{srv} is the average service time per message. This equation

illustrates that even with sufficient link bandwidth, poor queue depth control or increased service latency can lead to queue backlog.

$$P_{order} = 1 - \frac{N_{reorder}}{N_{total}} \quad (4)$$

Equation (4) gives the order retention rate metric. $N_{reorder}$ represents the number of out-of-order messages detected, and N_{total} represents the total number of messages. The goal of the high-frequency trading chain is not to eliminate all out-of-order messages in all cases, but to limit the unrecoverable range of out-of-order messages within a certain time frame, and ultimately ensure the consistency of the visible order of the application.

$$J = w_1 L_{p99} + w_2 U_{cpu} + w_3 R_{loss} + w_4 (1 - P_{order}) \quad (5)$$

Equation (5) establishes the comprehensive objective function for parameter tuning. L_{p99} represents the 99th percentile delay, U_{cpu} represents CPU usage, R_{loss} represents the loss or retransmission ratio, P_{order} represents the order retention rate, and w_i is the service weight. By adjusting the batch processing threshold, polling frequency, credit window, and standby QP switching threshold, the optimal solution in terms of delay, throughput, and reliability can be obtained.

Zero-copy data plane. To avoid frequent memory registration and release, the sender allocates a fixed-size message slot pool to each logical channel. The message body and metadata are registered as memory regions during initialization, and only the slot index is used during operation. For short messages of 64B-512B, inline descriptors and a compact payload layout are used to reduce cache line spanning; for larger messages, descriptors are used to store the offset and length, and remote writing is performed directly by the RNIC. The receiver uses a double-buffered reclamation mechanism to implement the function of only reclamping slots without data migration after the consumer thread completes its business, making the software path a simple process of writing descriptors, doorbell, polling completion, and index reclamation.

Providing an ordered submission mechanism for high-frequency trading. Since market data and order tracking are highly sensitive to order, this paper avoids a completely out-of-order approach. Instead, each channel stores a monotonically increasing sequence number and a submission threshold. The receiving end only allows out-of-order frames arriving within the window, but only sends frames to the upper layer if there are no missing frames to fill the gap and the backoff amount is excessive. To prevent duplicate visibility due to retransmissions, the system uses a message sequence number composed of a sequence number, channel number, and round number to identify messages, and performs idempotency checks during the submission phase.

Control plane and reliability design. The control plane does not carry large data flow, but only performs credit backhaul, queue level notification, fault detection and backup chain activation. Referring to Flor's software control path idea [1], this paper retains retry, credit adjustment and congestion backoff at the software level and automatically adjusts the strategy according to the trading period. The actual data transmission and acquisition are still completed by hardware, thus avoiding the control logic from entering the main data transmission channel. For the dual-machine hot standby requirement in high-frequency trading, each important channel additionally reserves a backup QP. When the main QP experiences continuous completion timeouts or credit stagnation, the control plane performs a fast switch and immediately submits the level.

Engineering Implementation Recommendations. First, replace interrupts with busy polling to achieve lower jitter. Second, bind the sending thread, polling thread, and transaction core thread to NUMA to reduce cross-core migration. Third, separate control messages and data messages into different channels to avoid short control messages being blocked by large packets. Fourth, complete

connection warm-up and page locking before market opening to eliminate the cold start overhead of the first packet. Fifth, set different batch processing thresholds according to different trading periods; moderately merge batches during stable periods to improve efficiency, and lower the threshold during volatile periods to prioritize tail latency.

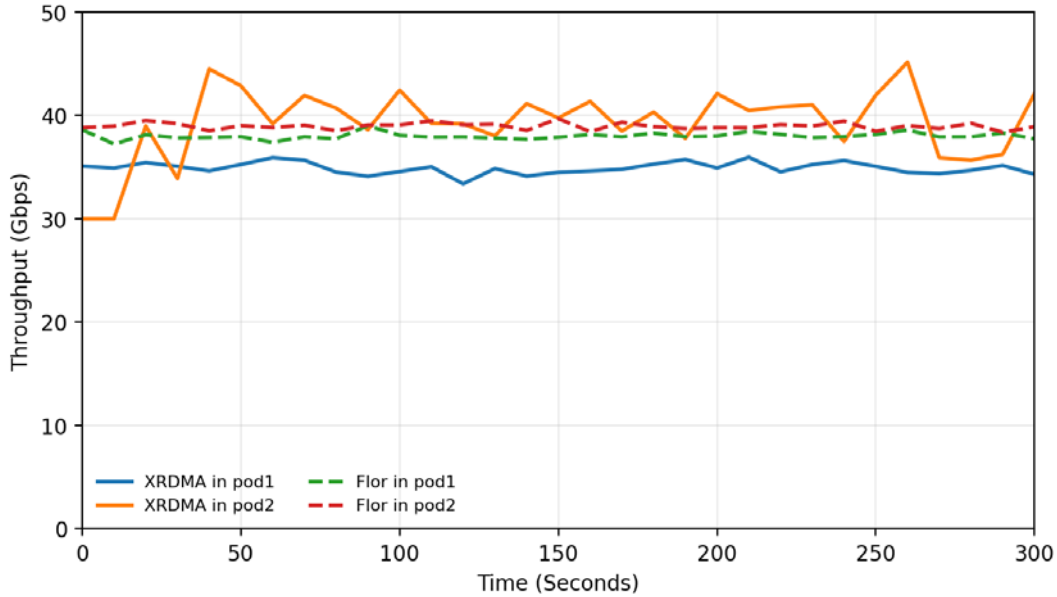


Figure 1 is a comparison of throughput fluctuations for various RDMA protocols in a cross-Pod scenario, drawn based on results published by Flor.

Figure 1 is a plot based on the experimental trends published in the Flor paper, with the horizontal axis representing time and the vertical axis representing throughput. All text in the figure is in English. As shown in Figure 3.3, under cross-pod conditions with lossy link disturbances, the throughput fluctuations between pods in traditional XRDMA are relatively large and unstable, exhibiting strong spatial inhomogeneity and jitter. However, the curves at both ends of the FLOR-type software control enhancement scheme are smoother, indicating that after the control plane perceives heterogeneous links and congestion feedback, the system stability is better guaranteed. For high-frequency trading, this result shows that without adjustable control plane support, zero-copy data planes will still suffer from tail latency instability under extreme market conditions.

Table 1 Different Middlewares Path and Characteristic

Path Type	Copies	Kernel Bypass	Ordering Scope	Tail-latency risk	HFT Suitability
TCP Broker	2-3	No	Broker-level	High	Low
Generic RDMA Verbs	0-1	Yes	Transport-level	Medium	Medium
Proposed RDMA ZC-MW	0	Yes	Channel-level	Low	High

Table 1 compares three typical paths in five aspects: number of copies, kernel bypass capability, scope of sequence control, tail latency risk, and adaptability to high-frequency trading. While traditional TCP brokers are relatively mature, their multiple copies and kernel involvement make them more suitable for general asynchronous services. General RDMA verbs already possess low overhead characteristics, but their sequence and business semantics still require custom encapsulation. The zero-copy message middleware proposed in this paper provides sequential commit and control plane governance at the channel level, thus better meeting the dual requirements of low jitter and business determinism in high-frequency trading.

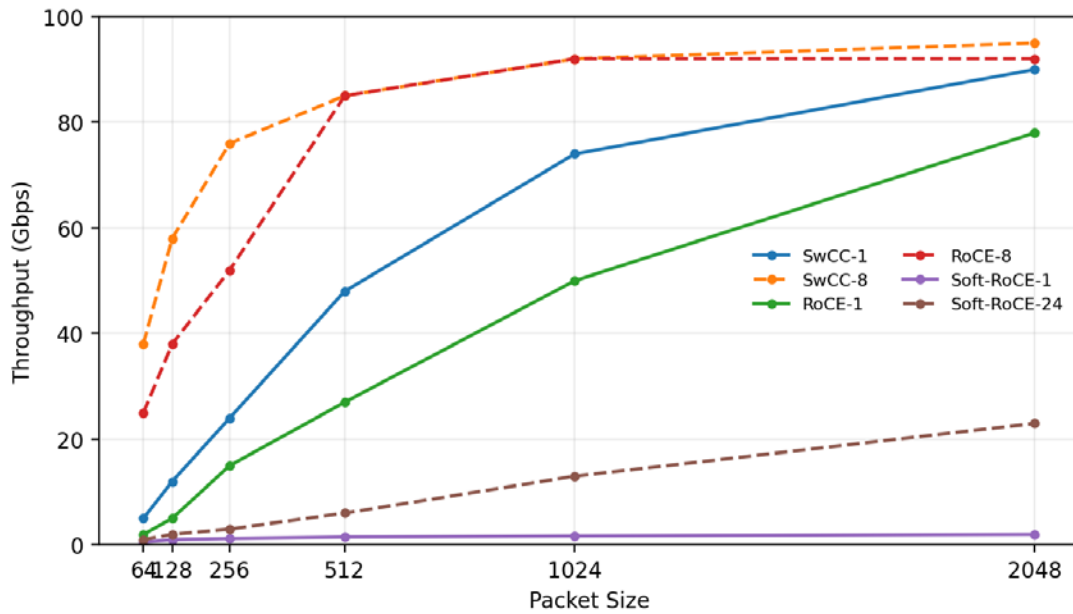


Figure 2 shows the redrawn graph of SwCC public results comparing throughput for different message lengths under different implementation methods.

Figure 2 is a reconstruction of the graphical data published in the SwCC paper, with the horizontal axis representing message length and the vertical axis representing throughput. The results show that as message size increases, the throughput of the hardware path or the hardware-software co-processing path quickly reaches line speed, while the pure software Soft-RoCE path grows slowly, especially for small message intervals. This issue is crucial for high-frequency trading because, when short messages dominate, any additional software processing step increases fixed overhead, thus eroding the data transmission profits that should belong to the RNIC. Therefore, middleware design should move business-irrelevant parts and protocol processing parts off the critical path as much as possible.

Table 2 Suggested tuning parameters for HFT-oriented RDMA middleware

Parameter	Low-volatility Session	Burst Session	Purpose	Expected Effect
Batch Threshold	8-16 msgs	1-4 msgs	Bound queueing delay	Lower p99 latency
Credit Window	64-128	32-64	Avoid sender overshoot	Reduce micro-burst risk
Polling Interval	Tight spin	Aggressive spin	Minimize completion jitter	Stable response time
Backup-QP Trigger	3 timeouts	1-2 timeouts	Fast failover	Higher availability

Table 2 provides parameter tuning recommendations for high-frequency trading. As shown in Figures 4-5, the configuration focus differs between stable and bursty periods. During stable periods, moderate batch processing can improve resource utilization, but care must be taken to avoid queuing latency accumulation. Parameter configuration is not aimed at increasing throughput; its purpose is to reduce tail latency, maintain sequentiality, and improve fault recovery capabilities. This is also a key difference between transactional middleware and general-purpose messaging systems.

5. Conclusion

This paper focuses on the design of a zero-copy message middleware based on RDMA for high-frequency trading scenarios. Building upon recent research on RDMA control programmability, connection management, reliability enhancement, and large-scale deployment, and considering the characteristics of high-frequency trading, a middleware architecture that can be implemented in an engineering context is proposed. The researchers believe that a truly suitable RDMA message middleware for high-frequency trading is not simply a verbs interface call, but rather one that reconstructs channel-level sequential semantics, fine-grained credit control, fast failover, and adaptive parameter tuning on the zero-copy data plane.

Methodologically, pre-registered memory pools, lock-free descriptor rings, busy polling completion queues, and control/data separation channels together form a prototype for reducing fixed overhead. From a business semantics perspective, bounded window reordering and sequential commit watermark mechanisms enable the system to maintain both hardware parallelism and transaction consistency. From an operational perspective, connection preheating, NUMA binding, backup QPs, and time-sharing parameter templates enhance the maintainability of the system in actual production environments.

Future research can explore three aspects: first, reducing CPU noise by offloading deeper control planes onto SmartNICs or DPUs; second, designing differentiated schedulers for three different types of traffic—market data, order placement, and risk control—to achieve different cross-link latency coordination goals; and third, using real transaction replay data to conduct more refined tail latency tests and anomaly injection experiments to verify the stability of the middleware under extreme market volatility. Based on RDMA zero-copy message middleware, a high-performance and deterministic communication infrastructure can be provided to high-frequency trading systems.

References

- [1] Wang, C. (2026). *Research on the Control of Uncertainty Risks in Investment Decision-making by Financial Modeling*.
- [2] Xinran Tu. *Resource Allocation Optimization and Cost Saving Analysis Based on Data Mining*. *International Journal of Business Management and Economics and Trade* (2026), Vol. 7, Issue 1: 1-9
- [3] Linwei Wu. *Data Visualization and Decision Support Analysis Based on Tableau*. *Socio-Economic Statistics Research* (2026), Vol. 7, Issue 1: 10-18
- [4] Yilin Fu. *Research on the Application of Innovative Financial Technologies in Capital Market Risk Management*. *Socio-Economic Statistics Research* (2026), Vol. 7, Issue 1: 1-9
- [5] Yuhan Zhou. *Green Bonds and Sustainable Financing Models in Energy Finance*. *International Journal of Social Sciences and Economic Management* (2026), Vol. 7, Issue 1: 28-35
- [6] Zhengle Wei. *Research on Innovative Design of Financial Derivatives and Market Risk Management Strategies*. *International Journal of Social Sciences and Economic Management* (2026), Vol. 7, Issue 1: 19-27
- [7] Shuang Yuan. *Research on Abnormal Detection and Transaction Risk Management Based on Machine Learning*. *International Journal of Social Sciences and Economic Management* (2026), Vol. 7, Issue 1: 10-18
- [8] Bukun Ren. *Multimodal Learning Method for Cross-Modal Data Alignment and Retrieval*. *International Journal of Multimedia Computing* (2026), Vol. 7, Issue 1: 1-8.

- [9] Yixian Jiang. *Performance Optimization and Improvement of Advertising Machine Learning Platform Based on Distributed Systems*. *International Journal of Big Data Intelligent Technology* (2026), Vol. 7, Issue 1: 9-17
- [10] Yiting Gu. *Application and Optimization Strategies of Cloud Services in Front end Engineering*. *International Journal of Big Data Intelligent Technology* (2026), Vol. 7, Issue 1: 1-8
- [11] Jin Li. *Performance Analysis of Efficient Microservice Architecture in the Financial Industry*. *Machine Learning Theory and Practice* (2026), Vol. 6, Issue 1: 1-9.
- [12] Qifeng Hu. *Optimization and Upgrade Path of Tax Management Software System Based on Cloud Platform*. *Socio-Economic Statistics Research* (2025), Vol. 6, Issue 2: 194-200.
- [13] Yilin Fu. *Data-driven Optimization of Capital Market Trading Strategies and Risk Management*. *International Journal of Business Management and Economics and Trade* (2025), Vol. 6, Issue 1: 221-228.
- [14] Linwei Wu. *Application and Development of Blockchain Technology in Financial Infrastructure Innovation*. *International Journal of Business Management and Economics and Trade* (2025), Vol. 6, Issue 1: 213-220.
- [15] Linwei Wu. *The Application of Quantitative Methods in Project Management and Actual Effect Analysis*. *International Journal of Business Management and Economics and Trade* (2025), Vol. 6, Issue 1: 204-212.
- [16] Yuanjing Guo. *The Practical Impact of an International Perspective on Promoting Financial Education*. *International Journal of Business Management and Economics and Trade* (2025), Vol. 6, Issue 1: 196-203.
- [17] Huijie Pan. *Discussion on Low-Latency Computing Strategies in Real-Time Hardware Generation*. *International Journal of Neural Network* (2025), Vol. 4, Issue 1: 57-64.
- [18] Wu Y. *Software Engineering Practice of Microservice Architecture in Full Stack Development: From Architecture Design to Performance Optimization*[J]. 2025.
- [19] Liu, D., Shen, Q., & Liu, J. (2026). *The Health-Wealth Gradient in Labor Markets: Integrating Health, Insurance, and Social Metrics to Predict Employment Density*.
- [20] Lu, Z. (2025). *Design and Practice of AI Intelligent Mentor System for DevOps Education*. *European Journal of Education Science*, 1(3), 25-31.
- [21] Zheng, H. (2025). *Research on Lifecycle Configuration and Reclamation Strategies for Edge Nodes Based on Microservice Architectures*. *Advances in Computer and Communication*, 6(5).
- [22] Zheng, H. (2025). *Research on Delay-aware Scheduling Algorithms for Edge Task Migration in High-concurrency Environments*. *Engineering Advances*, 5(4).
- [23] Hui, X. (2026). *Research on the Design and Optimization of Automated Data Collection and Visual Dashboard in the Medical Industry*. *Journal of Computer, Signal, and System Research*, 3(1), 27-34.
- [24] Shen, D. (2026). *Application of Large Language Model in Mental Health Clinical Decision Support System*. *International Journal of Engineering Advances*, 3(1), 23-30.
- [25] Wang, Y. (2026). *Research on Optimization of Neuromuscular Rehabilitation Program Based on Physiological Assessment*. *European Journal of AI, Computing & Informatics*, 2(1), 21-30.
- [26] Ding, J. (2026). *Optimization Strategies for Supply Chain Management and Quality Control in the Automotive Manufacturing Industry*. *Strategic Management Insights*, 3(1), 17-23.
- [27] Zhang, Q. (2026). *How to Improve Marketing Efficiency and Precision through AI-Driven Innovative Products*. *Strategic Management Insights*, 3(1), 1-8.
- [28] Liu, Y. (2026). *The Promoting Role of Fintech and Product Innovation in the Context of the Digital Economy*. *Strategic Management Insights*, 3(1), 9-16.
- [29] Shen, D. (2024, November). *Application of Machine Learning Algorithms Based on Magnetic Resonance Imaging in the Diagnosis of Knee Joint PVNS*. In *International Conference on*

Cognitive based Information Processing and Applications (pp. 401-411). Singapore: Springer Nature Singapore.

- [30] Lu, C. (2026). *Research on 3D Reconstruction Methods of Remote Sensing Images Combined with Deep Learning and GIS*. *International Journal of Engineering Advances*, 3(1), 15-22.
- [31] Cai, Y. (2026). *Design and Implementation of System Extensibility under High Concurrency Environment*. *International Journal of Engineering Advances*, 3(1), 31-37.
- [32] Liu, Y. (2026). *The Application of Data-Driven Financial Risk Management in Multinational Enterprises*. *Economics and Management Innovation*, 3(1), 20-26.
- [33] Huang, J. (2026). *Practice of Public Space Optimization and Functional Enhancement in Cultural Architecture*. *European Journal of Engineering and Technologies*, 2(1), 9-21.
- [34] Xu, D. (2026). *AI-Driven Video Content Optimization Strategies for Immersive Media*. *European Journal of Engineering and Technologies*, 2(1), 1-8.
- [35] Qi, Y. (2026). *AI Driven Payment System Security Improvement and User Privacy Protection Mechanism*. *Journal of Computer, Signal, and System Research*, 3(1), 35-41.
- [36] Qi, Y. (2026). *High Reliability Architecture and Compliance Design of Enterprise Level Financial Infrastructure*. *International Journal of Engineering Advances*, 3(1), 8-14.
- [37] Han, X. (2026). *Research on Automotive Manufacturing Process Optimization Methods for Multi-Supplier Collaboration*.
- [38] Dingyuan Liu. *Measuring the Sensitivity of Local Skill Structures to AI Substitution Risks Based on Occupational Task Decomposition*. *Socio-Economic Statistics Research* (2025), Vol. 6, Issue 2: 177-184
- [39] Wu, L. (2025, December). *Design and Application of Automatic Data Set Generation Tool Based on KLEE in Embedded Memory Management Performance Test Framework*. In *2025 IEEE 17th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 1111-1117). IEEE.
- [40] Wu, L. (2025, December). *Design and Application of Automatic Data Set Generation Tool Based on KLEE in Embedded Memory Management Performance Test Framework*. In *2025 IEEE 17th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 1111-1117). IEEE.
- [41] Truong, T. H. (2026). *Research on Risks and Countermeasures of Enterprise E-Commerce Transformation in the Cross-Border E-Commerce Environment*.
- [42] Qi, Y. (2025, October). *Research on Privacy Protection of AI Models in Big Data Using Differential Privacy Technology*. In *2025 2nd International Conference on Software, Systems and Information Technology (SSITCON)* (pp. 1-5). IEEE.
- [43] Hua, X. (2024, November). *Design and Implementation of a Game QoE Monitoring and Evaluation System Driven by Network Traffic Analysis*. In *International Conference on Cognitive based Information Processing and Applications* (pp. 149-161). Singapore: Springer Nature Singapore.
- [44] Huang, J. (2025, August). *Research on Multi-Model Fusion Machine Learning Demand Intelligent Forecasting System in Cloud Computing Environment*. In *2025 2nd International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)* (pp. 1-7). IEEE.