Scholar Publishing Group
Machine Learning Theory and Practice
https://doi.org/10.38007/ML.2026.060106
ISSN 2790-0983 Vol. 6, Issue 1: 49-58

Research on Monte Carlo Simulation Implementation of Option Pricing Based on FPGA Hardware Acceleration Engine for Low-Latency Derivatives Computation

Enqi Weng

School of Economics, Wuhan Donghu University, Wuhan, 430000, Hubei, P.R. China

Keywords: FPGA; option pricing; Monte Carlo simulation; hardware acceleration; low-latency computing

Abstract: To address the challenges in option pricing caused by the large sample size, latency sensitivity, and power consumption limitations of Monte Carlo simulations, this paper conducts in-depth research on four aspects: algorithms, random numbers, parallel pipelining, and resource mapping, and proposes a hardware acceleration engine framework based on FPGA. After reviewing the research progress of FPGAs in option pricing, quantitative finance, and random number generation over the past three years, the paper proposes an architecture goal for low-latency scenarios. This architecture focuses on three main aspects: geometric Brownian motion path generation, discounted return accumulation, and statistical error control, implemented using streaming data paths, path-level parallelism, on-chip random number supply, and fixed-point/floating-point co-design. The paper then presents the pricing model, discretization formula, throughput and energy efficiency evaluation indicators, and discusses the implementation methods from four aspects: module partitioning, pipelining scheduling, memory access organization, and precision control. By redrawing statistical graphs using the latest publicly available research data and conducting comparative analysis, it is shown that Versal-type AIE/PL co-design has significant advantages in throughput and energy consumption during large-scale path simulations. However, its performance ultimately depends on factors such as on-chip interconnects, random number generation bandwidth, and host-device coupling methods. Research findings indicate that the key to using FPGAs for Monte Carlo simulation of option prices lies not in parallelism, but in integrating random number generation, path movement, payout accumulation, and error control into a single pipeline using a reusable hardware engine.

1. Introduction

In financial computing tasks such as high-frequency risk measurement, market making, and intraday calibration, option pricing often requires a large number of repeated valuations to be completed in a very short time. For complex contracts corresponding to barrier options, basket options, and stochastic volatility models, analytical solutions are difficult to obtain. Therefore,

Copyright: © 2026 by the authors. This is an Open Access article distributed under the Creative Commons Attribution License (CC BY 4. 0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (https://creativecommons. org/licenses/by/4. 0/).

Monte Carlo simulation has become one of the most robust methods in engineering due to its good model adaptability and ability to estimate errors. At the same time, the statistical convergence speed of the Monte Carlo method is only O(N^-1/2). To achieve acceptable accuracy under millisecond-level or even lower latency budgets, the path throughput must be significantly improved. Recent reviews suggest that FPGAs have strong parallelism and high energy efficiency for option pricing, making them suitable for latency-sensitive financial work [1]. On the other hand, new devices such as Versal bring together programmable logic, on-chip networks, and AI Engines, leaving room for new schemes for financial simulation with fine-grained vector computation and streaming control [2-4].

However, hardware-based option pricing is not simply a matter of porting CPU code directly to an FPGA. Bottlenecks arise from path generation, random number transmission, on-chip cache organization, pipeline imbalance, and data coupling between the host and accelerator. Especially in the Monte Carlo framework, the quality and supply rate of random numbers directly determine the path progression speed. Recent research on FPGA random number generators shows that high-throughput true or pseudo-random modules capable of outputting stable bitstreams can meet the requirements within a small area, but integrating these modules with the mainstream financial simulation pipeline remains a major engineering challenge [5-7]. Therefore, establishing a dedicated hardware acceleration engine for option pricing has both theoretical research value and practical needs in trading systems for "low latency, interpretability, low power consumption, and reusability."

2. Analysis of the current status of the research topic

In terms of algorithms, the numerical calculation methods for option pricing mainly include the finite difference method, tree model method, Fourier transform method and Monte Carlo method. The first two methods have obvious advantages in low-dimensional European contracts, but when it comes to multi-asset, path-dependent or random volatility settings, the complexity of the curse of dimensionality and boundary handling will increase rapidly. In contrast, the Monte Carlo method can support the pricing of complex derivatives such as European, Asian, barrier and basket options with a unified path simulation platform. A 2024 review showed that FPGA achieved significant speed and energy efficiency gains in the Black-Scholes, tree model and Monte Carlo paths, among which the Monte Carlo route best fits the characteristics of hardware pipelining and parallelization [1].

In terms of hardware platform development, traditional FPGA acceleration is mostly driven by programmable logic and simplified reduction. Brown's research on Versal AI Engines in 2023 showed that AIE is suitable for handling rule vector operations, but the final performance is still affected by PL-AIE interaction bandwidth, window communication method and graph-level scheduling [2]. In 2015, Klaisoongnoen and Brown further explained the value of FPGA in low-power financial modeling from the perspective of quantitative financial applications, believing that in addition to high-frequency trading, risk analysis and derivatives calculation are also important application directions of FPGA. By 2024, Klaisoongnoen et al. conducted AIE/PL collaborative experiments on STAC-A2 derived workloads. The results showed that simply migrating the original PL design to AIE could not achieve optimal performance. The system bottleneck was more due to interconnection and division of labor than insufficient arithmetic units.

Financial Monte Carlo has put forward four requirements for random number generators: sufficient quality, sufficient speed, parallel independence, and controllable resources. In 2024, Chen et al. designed a high-speed dual-entropy source TRNG using FPGA and improved its random source speed and stability[5]. Ahmed et al. proposed using contention hazards and jitter

characteristics to generate a high-speed TRNG in IEEE Access, indicating that the random number module is gradually becoming a bottleneck control method at the system level. In 2025, Bouaziz and Fahmy gave the PRNGine scheme for AMDAI Engines, which deeply embedded pseudo-random number generation and distribution approximation into the data flow environment, providing financial Monte Carlo with a random number foundation that is more in line with pipeline requirements. At the same time, in 2025, Haas and Giles integrated nested MCMC with low-precision FPGA, showing that with the pursuit of low latency, precision-cost coordination improvement has also become a new stage[8].

Existing research has shown that FPGAs can achieve high throughput and high energy efficiency in financial numerical computation. Bouaziz et al.'s AIE data flow overlay design has obvious advantages over traditional PL and multi-threaded CPUs [9]. However, there are still three gaps in the engineering of low-latency weighted pricing. First, the algorithm model and hardware pipeline have not yet formed a stable mapping template. Second, the random number supply and path advancement are often separated from each other in the design, resulting in a large loss of system-level throughput. Third, most performance evaluations only focus on single-item speed and rarely consider the limits of latency, resources, energy consumption and error at the same time. Hajduk's simple TRNG design in 2025 shows that there is still a lot of room for optimization in the combination of low-cost random source and dedicated computing engine [10].

2.1 Second, raise the question.

Based on the current research, this paper mainly studies the following issues: how to construct a low-latency, engineering-reusable hardware accelerator for FPGAs in option pricing Monte Carlo simulations. Therefore, it can be divided into four aspects. First, at the model level, it is necessary to determine expressions suitable for hardware pipelined price evolution and payoff calculation, so that path generation, payoff calculation, and reduction statistics form a stable and regular data flow. Second, at the architecture level, it is necessary to solve the mismatch between the random number generation rate and the path advancement rate, preventing the high-performance computing array from "starving" due to random sources. Third, it is necessary to rationally allocate storage among on-chip BRAM/URAM, registers, and external storage, achieving localization in the access of state variables and parameters to reduce memory access bubbles. Fourth, at the evaluation level, it is necessary to establish a comprehensive evaluation system considering various indicators such as price accuracy, confidence interval, throughput, speedup ratio, and energy efficiency, and not only look at peak performance while ignoring the reliability of financial calculations.

Therefore, the problem this paper aims to address is not whether FPGAs can accelerate option pricing, but rather how to organize Monte Carlo option pricing into a pipelining, parallelizing, and scalable hardware engine suitable for FPGAs. To address this problem, this paper selects the representative geometric Brownian motion European option pricing process as the basic object, leaving interfaces for its subsequent extension to barrier options, multi-asset basket options, and the MLMC framework.

3. Problem Solving and Implementation Strategies

To address the above issues, this paper proposes a comprehensive approach that utilizes financial model standardization, random number modularization, path-driven pipelined implementation, and statistical reduction streaming. Using the evolution of stock prices under a risk-neutral measure as an example, asset prices follow a geometric Brownian motion:

$$dS_t = rS_t\,dt + \sigma S_t\,dW_t \tag{1}$$

The asset price at time S_t, the risk-free interest rate r, the volatility σ, and the standard Brownian motion W_t. This expression forms the unified stochastic differential equation basis for hardware path advancement.

During discretization, dividing the time limit T into M time steps Δt = T/M yields the single-path update formula.

$$S_{(k+1)} = S_k \exp[(r - 0.5\sigma^2)\Delta t + \sigma\sqrt{\Delta t} \cdot Z_k] \quad (2)$$

Zk refers to independent random numbers, which are based on the standard normal distribution within the standard normal distribution. For FPGAs, this formula can be broken down into a fixed pipeline stage: "drift term pre-computation—random perturbation generation—exponential operation—state update". To reduce path-level latency, $(r-0.5\sigma^2)\Delta t$ and $\sigma\sqrt{\Delta t}$ are often pre-programmed into parameter registers, and then vectorized random number inputs are used to drive multiple paths to proceed in parallel.

Taking a European call option as an example, its Monte Carlo estimate is:

$$C = e^{(-rT)} \cdot (1/N) \Sigma_{(i=1)}^{N} \max(S_T^{(i)} - K, 0) \quad (3)$$

Where N is the number of simulated paths, K is the execution price, and $S_T^{(i)}$ is the expiration price of the i-th path. This formula shows that the revenue function is executed only once at the end of the path. Therefore, in the hardware, a three-level structure of path advancement array, end-of-path revenue calculator, and tree reducer can be used to directly accumulate the revenue results of many paths into the on-chip statistics module.

Considering that Monte Carlo estimations need to provide statistical reliability, this paper uses standard error constraints to ensure this.

$$SE = \sqrt{[(1/(N(N-1))) \Sigma_{(i=1)}^{N} (X_i - \bar{X})^2]} \quad (4)$$

The payoff for a single path is discounted by the discount rate, and the mean is denoted by $\bar{X}$. In hardware implementation, the mean and variance can be output using online variance updates or block reduction, so that the system provides an error estimate in addition to the price, thus meeting the needs of financial computing for a reliable interval.

In terms of performance evaluation, this paper uses speedup ratio as the core hardware metric:

$$Speedup = T_{CPU} / T_{FPGA} \quad (5)$$

This is combined with unit energy consumption throughput and latency statistics for a comprehensive evaluation. If Throughput is further defined as N/T_exec, different architectures can be compared, and the suitability for real-time calls on the transaction side can be determined from the perspective of EnergyEff=Samples/J.

The hardware engine developed in this paper consists of five modules: a parameter loading module, a random number module, a path advancement module, a revenue and reduction module, and a control and result feedback module. Unlike the traditional model where software dominates and hardware performs the kernel functions, this paper proposes integrating random number generation, path advancement, and statistical induction into a single data flow system, allowing the host computer to focus solely on task issuance and result acquisition.

From an implementation perspective, path-level parallelism should be used instead of time-level parallelism. Because the time dimension has recursive dependencies, expanding computations along the time dimension would lead to resource waste; while paths are naturally independent, making them more suitable for replicated computations. High-cost operations such as exponentiation and square root should use lookup tables, piecewise approximation, or DSP-friendly approximation

kernels to balance accuracy and area. Furthermore, for the AIE/PL collaborative platform, regular vector arithmetic and high-throughput random number transformations should be preferentially mapped to AIE, while state caching, control synchronization, and complex reduction should be placed on PL to reduce cross-domain round trips. Finally, the accuracy strategy can adopt a hybrid approach of "maintaining high accuracy for critical accumulation and moderately low accuracy for path advancement," which aligns with the 2025 MLMC on FPGA concept—sacrificing small errors for limited accuracy in exchange for significant cost reduction.

4. Experimental Comparison and Discussion of Results

The research objective of this paper is to propose a feasible FPGA hardware engine solution. The paper uses the latest publicly available data to redraw and compare the rationality and feasibility of "streaming parallelism," "random number collaboration," and "hierarchical resource mapping." Figure 1, redrawn based on data from Table 3 published in 2024, shows the total execution time of different platforms for different problem sizes. Figure 2, redrawn using energy efficiency measurement results published in 2025, compares the power consumption and energy consumption of the GPU and AIE solutions.

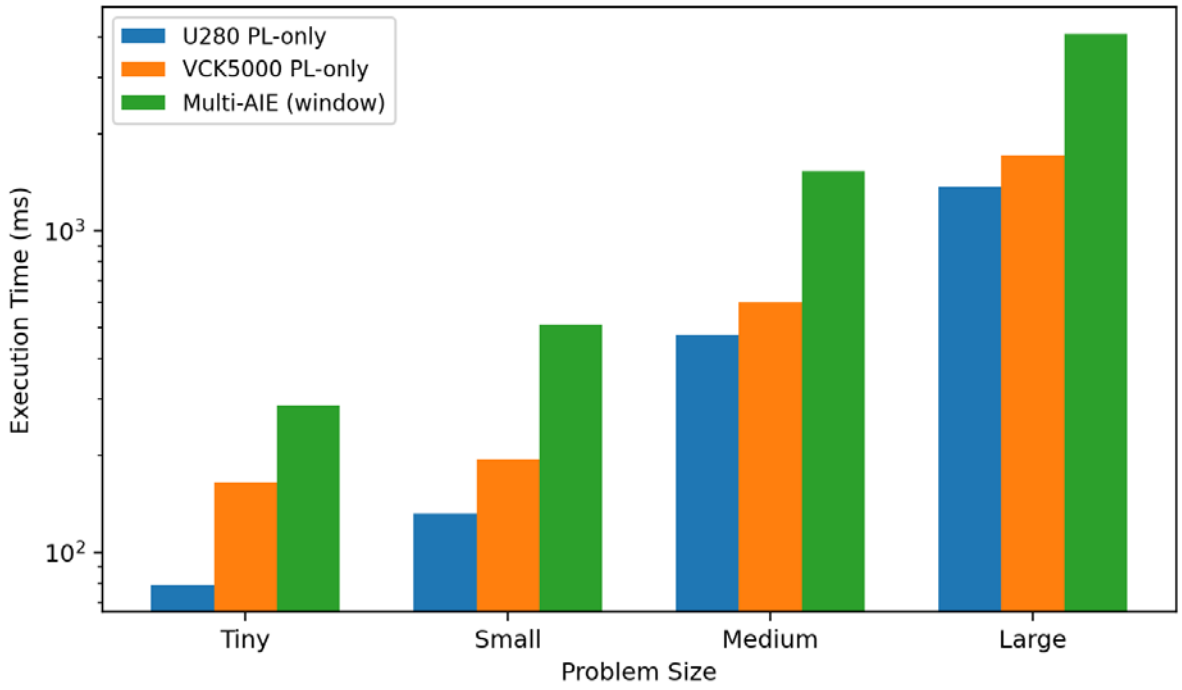

Figure 1. Comparison of execution times for different problem sizes

Figure 1 uses the logarithmic vertical axis to illustrate the execution time for four problem sizes: Tiny, Small, Medium, and Large. Figure 3 shows that the traditional U280PL only exhibits good absolute latency across all problem sizes. However, when using the PL on the VCK5000, the overall latency is actually higher due to DRAM constraints and architectural differences. While the absolute latency of the Multi-AIE (window) scheme is not yet completely superior to the optimal PL-only design, it demonstrates stronger scalability potential compared to the unoptimized VCK5000 PL-only design. The performance improvement on the new FPGA platform is not due to device upgrades, but rather to the partitioning of computing units, window communication methods, and PL/AIE resource coordination.

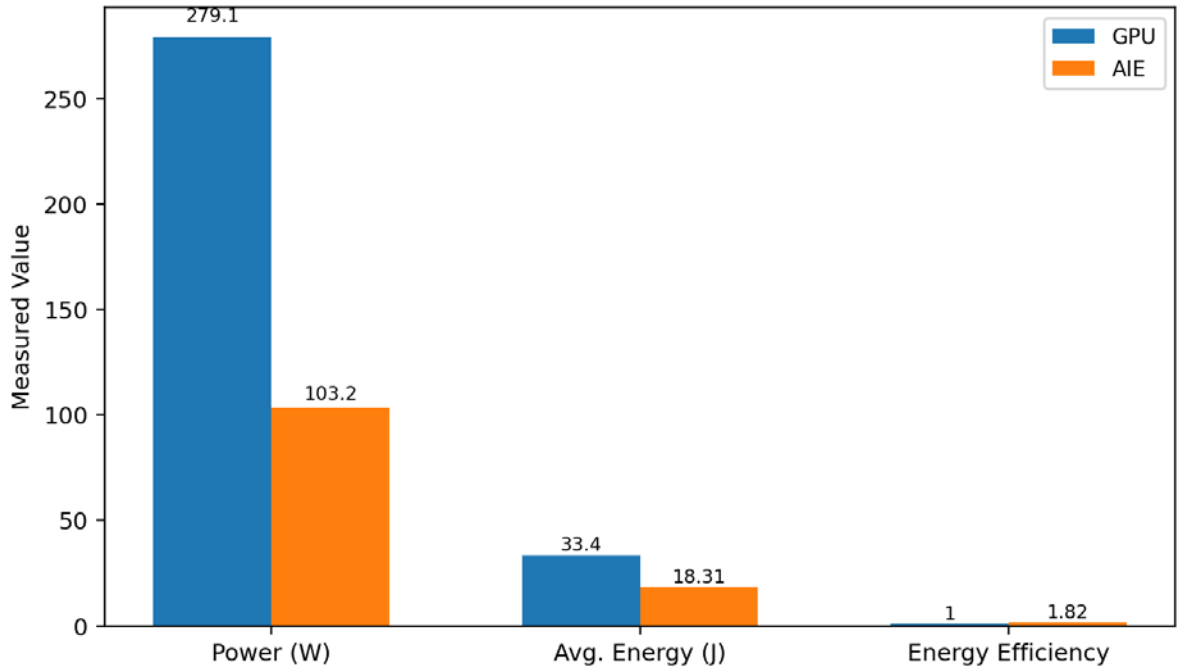

Figure 2. Comparison of power consumption and energy consumption between GPU and AIE solutions.

Figure 2 compares the power consumption, average energy consumption, and energy efficiency of GPUs and AIE accelerators. Although GPUs are faster at a certain point in time, the average energy consumption using the AIE solution is only 18.31 J, which is 95.2% faster than GPUs, and its power consumption is only 1.82 times that of GPUs. For trading and risk control systems that need to perform pricing tasks repeatedly, the speed achievable per unit time is important, but the energy consumed in continuous operation and the deployment density are equally critical. The results show that FPGA-based platforms for financial infrastructure have better long-term operational economics.

Table 1: Proposed fpga Engine Config For Montecarlo Option pricing

Module	Function	Parallelism	Implementation Note
Param Loader	Load model constants	Task-level	Keep drift and diffusion coefficients on-chip
RNG Core	Uniform/Normal samples	Path-level	Support parallel substreams and vector output
Path Engine	GBM path update	Pipeline + Path-level	Map exp/mul/add kernels to DSP-friendly stages
Payoff Reducer	Discounted payoff and sum	Tree reduction	Output mean and variance online
Host Interface	Task dispatch and return	Burst transfer	Minimize control traffic during steady state

When proposing the implementation framework, Table 1 shows the engine module division for Monte Carlo simulation of European options; Table 2 shows the main results of representative studies in the past three years.

Table 1 summarizes the hardware engine architecture proposed in this paper from an engineering

implementation perspective. Its main idea is to decouple parameter loading, random number provision, path advancement, revenue calculation, and host interaction into five modules that can be optimized independently. Among them, the RNG Core and Path Engine directly determine the system's peak throughput, while the Payoff Reducer relates to whether statistical errors can be output online. This modular division is applicable to both replicative expansion on traditional PLs and mapping vector arithmetic to AIEs and control and caching to PLs on the Versal platform, forming a clear hardware-software collaborative boundary.

Table 2: Some latest representative study about FPGA option pricing & support module

Year	Study	Platform	Focus	Key Result
2023	Brown	Versal AIE	AIE/PL coupling	AIE scaling depends on communication pattern
2024	Klaisoongnoen et al.	Versal VCK5000	Risk analysis kernel	AIE version still bottlenecked by PL interfaces
2024	Chen et al.	FPGA TRNG	Random source	High-throughput dual-entropy source for FPGA
2025	Haas and Giles	FPGA + MLMC	Precision-cost trade-off	Nested MLMC reduces cost with tuned precision
2025	Bouaziz et al.	AMD Versal AIEs	Multi-asset pricing	Up to 25.7x over PL library baseline

Table 2 shows that in recent years, research has shifted from verifying whether FPGAs can accelerate financial computing to organizing accelerators in a systematic way. In 2023, more discussion focused on the programmability and collaborative boundaries of new platforms. In 2024, attention began to be paid to the bottlenecks encountered by specific financial kernel functions on Versal, with improved throughput of the random number module being a supporting technology. By 2025, more complex scenarios such as multi-asset options and MLMCs began to enter the mainstream of FPGA research, indicating that the industry's focus has shifted from single-operator optimization to the scalable design of the entire financial computing pipeline. The implementation strategy in this paper is consistent with this development trend.

Further analysis reveals that three principles must be followed for low-latency FPGA option pricing engines. First, the random number bandwidth must be determined before the computing array planning; otherwise, even with numerous path units, effective throughput cannot be achieved. Second, revenue reduction and error estimation cannot be performed on the host side; otherwise, PCIe or on-chip interface backhaul will negate the benefits of hardware acceleration. Third, performance evaluation should consider multiple aspects such as latency, path throughput, resource consumption, and energy efficiency. In actual trading systems, continuous power consumption and heat dissipation directly affect the deployment density of the data center. If barrier options or basket options are subsequently developed in the proposed solution, upgrades can be achieved by adding path state registers and condition judgment logic to the Path Engine, and adding correlation matrices or boundary monitoring units to the parameter side, without changing the overall framework.

5. Conclusion

This paper, titled "Implementation of an FPGA-Based Hardware Acceleration Engine in Monte Carlo Simulation of Option Pricing," reviews relevant research from recent years and proposes an implementation framework suitable for low-latency financial computing scenarios. Research reveals that the key to hardware-based Monte Carlo option pricing lies in treating random number generation, path progression, payoff aggregation, and error control as a continuous data stream,

rather than improving the kernel function at a single stage in isolation. Model formulas, system architecture, and publicly available experimental data demonstrate that FPGAs offer advantages over other types of devices in terms of throughput per unit of energy consumption; however, final performance is limited by random number supply capacity, on-chip interconnection methods, and the coupling efficiency between the host and the device. Therefore, the modular engine structure proposed in this paper has good scalability and engineering feasibility, and can be further extended to more complex scenarios such as multi-asset options and barrier options. Future work can focus on three aspects: first, using hierarchical random number generators and parallel subsequence management strategies; second, researching methods to reduce area and power consumption, such as mixed precision and approximate exponential kernels; and third, creating a software-hardware co-optimization closed loop oriented towards business SLAs based on the latency budget of the real trading side.

References

[1] *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*

[2] *Han, X. (2026). Research on Automotive Manufacturing Process Optimization Methods for Multi-Supplier Collaboration.*

[3] *Han, X. (2026). Research on Process Decision-Making Behavior under Incomplete Information Conditions in Automobile Manufacturing Systems.*

[4] *Hou, Y. (2026). Research on Server Performance Stability Assurance Mechanisms during Cross-Generation Computing Platform Upgrades.*

[5] *Hou, Y. (2026). Research on BIOS and BMC Compatibility Optimization Methods for Cross-Generation Servers in Production Environments.*

[6] *Yin, J. (2026). Research on Financial Time Series Prediction and Multiscale Correlation Based on the Fusion of Network Big Data and Deep Learning.*

[7] *Yu, X. (2026). Strategy Models and Practical Research of Growth Marketing under the Background of Digital Transformation.*

[8] *Yu, X. (2025). Digital Transformation Empowers Growth Marketing with Marketing Data Analysis Integration and Real-Time Display Strategy.*

[9] *Liu, H. (2026). Research on Dynamic Price Prediction of E-commerce Based on Time Series Modeling.*

[10] *Liu, H. (2026). Research on the Application of Causal Reasoning Method in Content Compliance Experimental Evaluation.*

[11] *Wang, Y. (2026). Research on the Application of Artificial Intelligence in Supply Chain Risk Early Warning.*

[12] *Sun, Q. (2026). Research on a Robotic Natural Language Intelligent Decision-Making Framework Based on Large Language Models, Thinking Chain Reasoning, and Multi-Agent Collaboration.*

[13] *Sun, Q. (2026). Research on Lightweight Intelligent Dialogue Systems Based on Semantic Entity Enhanced Intention Recognition and Rule Retrieval Generation Hybrid Models.*

[14] *Zhou, Y. (2024, November). Construction of a Multi-factor Quantitative Stock Selection System for the New Energy Industry Based on Microservices Architecture and Machine Learning Components. In International Conference on Cognitive based Information Processing and Applications (pp. 163-174). Singapore: Springer Nature Singapore.*

[15] Huang, J. (2025, September). Performance Evaluation Index System and Engineering Best Practice of Production-Level Time Series Machine Learning System. In 2025 International Conference on Intelligent Communication Networks and Computational Techniques (ICICNCT) (pp. 01-07). IEEE.

[16] 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.

[17] 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.

[18] 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.

[19] Truong, T. H. (2026). Research on Risks and Countermeasures of Enterprise E-Commerce Transformation in the Cross-Border E-Commerce Environment.

[20] 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.

[21] 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.

[22] Chen, X. (2024, November). Cloud Storage User Behavior Analysis and Dynamic Replica Strategy Optimization Based on Improved RFM and Fuzzy Clustering. In International Conference on Cognitive based Information Processing and Applications (pp. 425-434). Singapore: Springer Nature Singapore.

[23] Wang, C. (2026). Research on the Control of Uncertainty Risks in Investment Decision-making by Financial Modeling.

[24] 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

[25] Linwei Wu. Data Visualization and Decision Support Analysis Based on Tableau. Socio-Economic Statistics Research (2026), Vol. 7, Issue 1: 10-18

[26] 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

[27] 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

[28] 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

[29] 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

[30] Lu, Z. (2025). AI-Driven Cross-Cloud Operations Language Standardisation and Knowledge Sharing System. European Journal of AI, Computing & Informatics, 1(4), 43-50.

[31] Bukun Ren. Multimodal Learning Method for Cross-Modal Data Alignment and Retrieval. International Journal of Multimedia Computing (2026), Vol. 7, Issue 1: 1-8.

[32] 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

[33] Jin Li. Performance Analysis of Efficient Microservice Architecture in the Financial Industry. Machine Learning Theory and Practice (2026), Vol. 6, Issue 1: 1-9.

[34] 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.

[35] Lu, Z. (2025). Design and Practice of AI Intelligent Mentor System for DevOps Education. European Journal of Education Science, 1(3), 25-31.

[36] Wang, C. (2025). Research on Market Evaluation Strategies for Financial Institutions Based on Big Data Analysis.

[37] Zhang, X. (2025). Optimization and Implementation of Time Series Dimensionality Reduction Anti-fraud Model Integrating PCA and LSTM under the Federated Learning Framework. Procedia Computer Science, 262, 992-1001.

[38] Xuanrui Zhang. Exploration of the Application of Big Data Technology in Financial Fraud Monitoring. Socio-Economic Statistics Research (2026), Vol. 7, Issue 1: 19-26

[39] Hong, Y. (2025). Architecture Design and Performance Optimization of a Large-scale Online Simulation Platform for Business Decision-making. Advances in Computer and Communication, 6(4).

[40] Hong, Y. (2026). Research on Warehouse Capacity Optimization Methods Based on Predictive Modeling. Engineering Advances, 6(1).

[41] Truong, T. H. (2025). Research on the Application of Digital Healthcare Platforms in Chronic Disease Management. Advances in Computer and Communication, 6(5).

[42] Xu, D. (2026). AI-Driven Video Content Optimization Strategies for Immersive Media. European Journal of Engineering and Technologies, 2(1), 1-8.

[43] Huang, J. (2026). Practice of Public Space Optimization and Functional Enhancement in Cultural Architecture. European Journal of Engineering and Technologies, 2(1), 9-21.

[44] Liu, Y. (2026). The Application of Data-Driven Financial Risk Management in Multinational Enterprises. Economics and Management Innovation, 3(1), 20-26.