MultiHedge: Adaptive Coordination via Retrieval-Augmented Control

Feliks Bańka[0009−0005−1973−5861]
and Jarosław A. Chudziak[0000−0003−4534−8652]

The Faculty of Electronics and Information Technology,
Warsaw University of Technology, Warsaw, Poland
{feliks.banka.stud, jaroslaw.chudziak}@pw.edu.pl

Abstract. Decision-making under changing conditions remains a fundamental challenge in many real-world systems. Existing approaches often fail to generalize across shifting regimes and exhibit unstable behavior under uncertainty. This raises the research question: can retrieval-augmented LLM coordination improve the robustness of modular decision pipelines? We propose MultiHedge[1], a hybrid architecture where an LLM produces structured allocation decisions conditioned on retrieved historical precedents, and execution is grounded in canonical option strategies. In a controlled evaluation using U.S. equities, we compare MultiHedge to rule-based and learning-based baselines. The key result is that memory-augmented retrieval confers greater robustness and stability than increasing model scale alone. Our paper contributes a controlled computational study showing that memory and architectural design play a central role in robustness in modular decision systems.

Keywords: Computational experiments · Retrieval-augmented systems · Large language models · Risk management

1 Introduction

Financial markets are characterized by frequent regime shifts and volatility bursts, complicating risk management and making risk management unstable [13]. Even established option strategies such as collars and straddles provide only partial protection if not coordinated over time [2]. Selecting and adjusting these hedges under changing market conditions is a sequential control problem involving both discrete and continuous decisions [12].

Recent advances in large language models (LLMs) have enabled structured workflows that integrate heterogeneous signals and produce auditable rationales [6]. Retrieval-based memory mechanisms offer a lightweight way to condition decisions on similar past states, improving robustness under distribution shift [1]. However, most hedging frameworks focus on end-to-end learning or

[1] Code and experimental configuration files are publicly available at: https://github.com/latent-systems-lab/MultiHedge

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39

static overlays, rarely studying retrieval-conditioned coordination over modular, interpretable decision primitives [11].

This raises a central research question: can retrieval-augmented coordination enhance the robustness of modular decision pipelines under distributional shift and regime instability? We address this by introducing *MultiHedge* (Section 3.1), a hybrid computational architecture in which the LLM serves as a bounded heuristic approximator within a constrained optimization loop. Allocation decisions are strictly conditioned on retrieved historical analogues, while execution is handled by formally defined option-hedging modules. This design dynamically coordinates canonical strategies, separating stochastic reasoning from symbolic execution to align language-model inference with constrained portfolio control [7].

The main contributions of this work are as follows. First, we formulate retrieval-conditioned coordination as a structured sequential decision problem. Second, we introduce a modular hybrid architecture that separates reasoning, action generation, and execution within a reproducible workflow. Third, we present a controlled computational study—including limited robustness checks—that isolates the role of memory and coordination in improving robustness beyond model scale.

2 Related work

Large language models (LLMs) and retrieval-augmented generation (RAG) workflows have been integrated into structured decision systems to ground outputs in external evidence and improve interpretability [5, 7]. In finance, such systems support sentiment extraction, event interpretation, and tool-augmented trading components [16]. Retrieved episodes provide precedents that align model reasoning with historical cases [1]. Conditioning decisions on retrieved experiences improves robustness under distribution shift and non-stationarity [15], with episodic memory acting as a non-parametric prior that grounds inference in observed trajectories rather than purely parametric knowledge. In structured decision architectures, LLMs increasingly operate as bounded reasoning modules over constrained action spaces, where outputs are parsed and executed by deterministic components [5, 10]. In *MultiHedge*, episodic recall informs allocation decisions over option-level primitives, coupling regime inference with historical outcome alignment [16].

From a computational-science perspective, modular control architectures are standard mechanisms for robustness under non-stationarity and partial observability [12, 8]. Specialist modules capture complementary behaviours, while a coordinator performs context-dependent selection under explicit constraints. Learning-based hedging and risk-control frameworks demonstrate that such policies can operate under frictions when embedded into a well-defined execution model [4], and classical parameterised primitives remain widely used due to predictable cost–benefit trade-offs [10]. Existing financial LLM systems such as FinAgent primarily emphasize signal interpretation and tool use [16]. Ensemble and mixing approaches improve downside stability [3, 14], yet they rarely formalise

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39

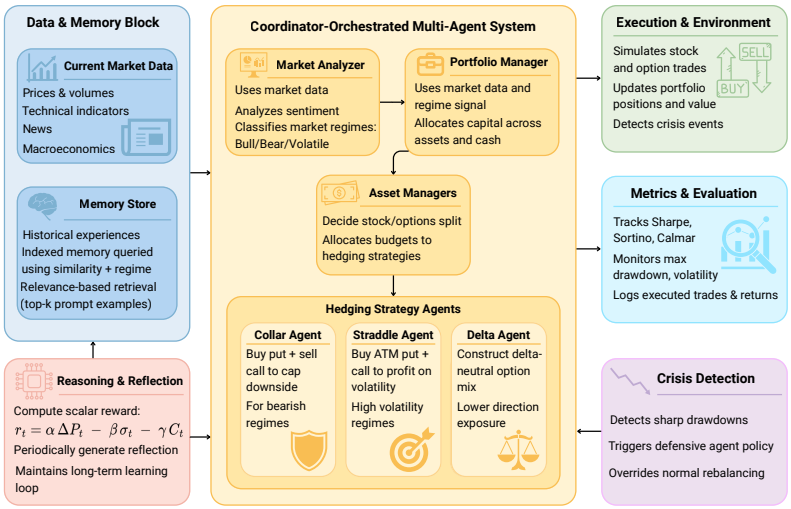

Fig. 1. MultiHedge system architecture: modular components, episodic memory, and hybrid control pipeline.

retrieval-conditioned coordination over fixed interpretable primitives within a constrained sequential control and backtesting framework. Our work addresses this gap by explicitly separating retrieval, allocation synthesis, and symbolic execution in a reproducible computational workflow.

3 MultiHedge: Hybrid Architecture, Episodic Memory, and Modular Coordination

This section details the MultiHedge architecture, which integrates episodic memory, modular coordination, and a hybrid control pipeline. Figure 1 illustrates the system's main components and their interactions.

We formulate dynamic hedging as a constrained sequential control problem under partial observability, where portfolio allocation must remain bounded under regime shifts and transaction frictions. Partial observability arises because regime and volatility states are latent.

Formally, the problem is represented as a Partially Observable Markov Decision Process (POMDP) [12]:

$$\langle \mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}, \mathcal{R} \rangle,$$

where $\mathcal{S}$ is the latent market–portfolio state space, $\mathcal{A}$ is a mixed discrete–continuous action space, $\mathcal{O}$ are observable signals (prices, volatility, news features), $\mathcal{T}$ denotes transition dynamics, and $\mathcal{R}$ is a risk-adjusted reward functional. *MultiHedge* implements a hybrid computational architecture that combines symbolic option-hedging structures, retrieval-augmented memory, and language-model-based reasoning within this decision process.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39

3.1 System Overview

As shown in Figure 1, MultiHedge consists of several interacting modules: an LLM-based inference layer, a portfolio and allocation layer, symbolic hedging agents, and a deterministic safety layer. The figure provides a visual summary of how these components coordinate to produce memory-augmented decisions.

The LLM processes multimodal signals and outputs regime classification and structured allocation candidates. The portfolio layer implements a risk-adjusted objective, while hedging agents execute canonical option strategies. The safety layer enforces stability constraints by overriding actions during drawdown events, ensuring bounded risk and robust system behavior.

3.2 Decision Process and Action Formulation

We model hedging as a constrained decision process where, at each trading day t, the system observes a partially informative state and produces structured allocation and hedging actions subject to feasibility and stability constraints.

Formally:

$$w_t = \pi_{\text{alloc}}(s_t, R_t, E_t), \tag{1}$$

$$a_{i,t}^{H} = \pi_{\text{agent}}^{H}(s_t, w_t), \quad \forall H \in \{\text{collar, straddle, delta-neutral}\}. \tag{2}$$

Under this formulation, the LLM acts as a constrained reasoning engine. It produces structured outputs that are grounded via retrieval and executed through interpretable primitives. These outputs are parsed and validated against explicit financial constraints, including budget, liquidity, and risk limits. This ensures that all actions remain feasible, interpretable, and bounded by the system's operational parameters at every step.

To mitigate hallucinations and enforce stability, a deterministic safety layer overrides the LLM's outputs when drawdown thresholds are breached. Deterministic inference (temperature zero) further reduces stochasticity, ensuring predictable behavior under stress.

The retrieval-augmented policy update is given by:

$$\pi^* \leftarrow \arg\max_{\pi} \; \mathbb{E}_{(s,a)\sim\mathcal{M}} \left[\text{sim}(\phi(s,a), \phi(s',a')) - \lambda\, \mathcal{L}(\pi)\right]$$

where $\text{sim}(\cdot,\cdot)$ denotes embedding similarity, ϕ is the embedding function, λ controls retrieval strength, and $\mathcal{L}(\pi)$ is a regularization term. Under this formulation, retrieval constrains policy updates toward historically aligned trajectories, acting as a non-parametric regularizer under distributional shift.

3.3 Memory-Augmented Reasoning via Metric Retrieval

As illustrated in Figure 2, the system follows a retrieval–reason–act–update loop driven by a non-parametric episodic memory buffer $\mathcal{M}$. Past decision episodes are stored within this buffer as state–action–outcome tuples, establishing a continuous

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39

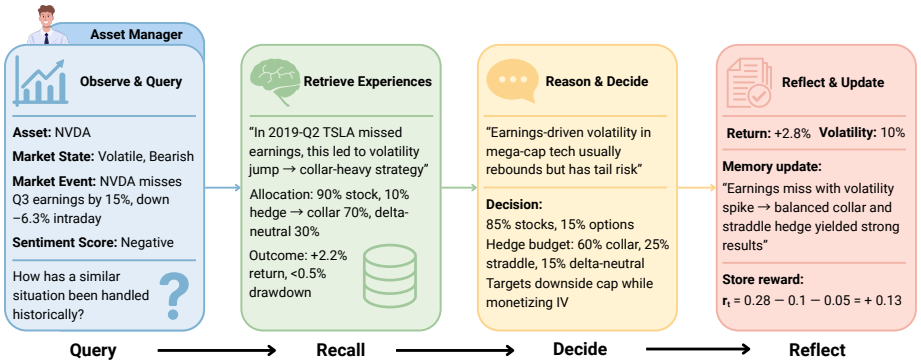

Fig. 2. Episodic memory loop in MultiHedge: past state–action–outcome tuples are retrieved via similarity search, used to condition current allocation decisions, and then stored for future reuse.

cycle where historical trajectories are retrieved to condition current actions and then updated for future reuse.

During the decision phase, the controller retrieves the top-k most similar episodes using cosine similarity and anchors its allocation logic to their realized outcomes. By explicitly conditioning on these retrieved precedents, the system reuses empirically validated strategies in analogous contexts rather than relying solely on parametric generalization.

4 Experimental Evaluation under Changing Market Conditions

We evaluate MultiHedge in a setting where market conditions vary over time. The system is compared against classical and learning-based baselines, and we perform targeted ablations to isolate the effects of memory, coordination, and model scale.

4.1 Experimental Setup and Robustness Metrics

We use daily historical market data for large-cap equities (AAPL, TSLA, NVDA) from January 2021 to December 2023, with memory index calibration on 2016–2020. Trading costs and liquidity constraints are enforced, and the LLM controller GPT-4.1 operates deterministically ($T = 0.0$).

Evaluation relies on core performance and tail-risk metrics: Total Return (TR), Sharpe Ratio (SR; return per unit of total volatility), Maximum Drawdown (MD; largest peak-to-trough drop), and Conditional Value at Risk (CVaR). We additionally track auxiliary stability indicators: Sortino (SoR) and Calmar (CaR) ratios for downside-adjusted returns, Value at Risk (VaR), worst-month return (WM), downside deviation (DDv; volatility of negative returns), time spent in

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39

Table 1. Performance Comparison: Core Metrics and Downside Risk

Strategy	SR	TR (%)	MD (%)	CVaR (%)	WM (%)	DDv (%)
Buy & Hold	0.81	107.08	53.08	5.36	-21.96	25.34
Equal Weight	0.77	95.34	53.53	5.41	-21.29	25.73
PPO	0.74	66.46	39.72	3.95	-15.59	18.68
FinAgent	0.63	69.74	54.60	5.99	-21.92	28.69
FinRL	0.56	57.08	61.51	6.48	-25.19	30.78
MultiHedge	**1.69**	**152.08**	**16.22**	**2.29**	**-4.58**	**10.93**

severe drawdown (TDD $> 10\%$), and maximum consecutive loss days (MCL). These metrics exhibit consistent trends and reinforce the observed improvements in risk-adjusted performance. Auxiliary robustness checks, including broad-market stress and synthetic regime-switching tests, further confirm the stability.

4.2 Benchmark Results

Table 1 shows that MultiHedge consistently improves both performance and risk characteristics relative to all baselines. The Sharpe ratio increases substantially compared to classical strategies, while total return improves without a corresponding increase in risk. The most pronounced gains appear in downside risk. Maximum drawdown is reduced by over 70% relative to Buy & Hold ($53.08\% \rightarrow 16.22\%$), and CVaR is more than halved. The worst-month loss improves from approximately -22% to -4.6%, indicating strong mitigation of extreme negative events. Similarly, downside deviation is reduced by more than 50%, suggesting that improvements are not limited to isolated tail events but reflect a broader compression of downside volatility.

These effects indicate that MultiHedge not only improves average performance but systematically reshapes the loss distribution, reducing both the magnitude and persistence of adverse regimes. Consistent improvements across additional monitored metrics (SoR, CaR, VaR, TDD, MCL) further support this interpretation. Auxiliary tests under stress and regime-switching scenarios confirm that these gains remain stable across perturbations, validating retrieval-conditioned coordination as a robust mechanism for non-stationary environments.

4.3 Architectural Ablation Studies

We conduct ablation studies to isolate the effects of memory, stochasticity, and model scale. GPT-4o Mini is used as a reduced-capacity baseline to assess the effect of model scale under fixed architecture. Table 2 shows that memory is the dominant factor driving robustness. Removing memory leads to a near tripling of drawdown ($16.22\% \rightarrow 46.18\%$) and a substantial deterioration in tail risk (CVaR: $2.29 \rightarrow 4.78$). The degradation is also visible in worst-month losses and downside deviation, indicating both more frequent and more severe adverse outcomes.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39

Table 2. Ablation of Control Architecture: Core and Downside Risk Metrics

Configuration	SR	TR (%)	MD (%)	CVaR (%)	WM (%)	DDv (%)
MultiHedge (GPT-4.1)	1.69	152.08	16.22	2.29	-4.58	10.93
GPT-4o Mini	1.14	97.44	20.41	2.87	-9.26	13.77
Stochastic ($T = 0.7$)	0.96	84.58	28.28	3.28	-12.27	15.86
No Memory	0.80	93.94	46.18	4.78	-19.95	22.89

Introducing stochasticity further degrades performance, suggesting that deterministic, constrained inference is important for stability in sequential decision settings. In contrast, reducing model capacity results in a comparatively smaller performance drop across all metrics. While we consider a limited comparison of model scales, these results indicate that architectural components—particularly memory—have a stronger impact on robustness than model size in this setting.

5 Discussion and Future Work

The empirical outperformance of MultiHedge over reinforcement learning baselines highlights a broader shift in designing computational financial agents. Deep RL approaches, such as PPO or FinRL [9], often struggle with sudden regime shifts because they optimize over historically stationary reward landscapes [4]. Conversely, purely LLM-based trading systems [16] risk unstable, unconstrained execution. By restricting the LLM to a coordinating role and offloading execution to deterministic primitives, our architecture achieves robust downside protection without requiring end-to-end retraining. This suggests that grounding modular components via episodic memory can provide a more stable path to robustness than relying solely on parametric scale [7, 5].

Despite these advantages, limitations remain. Restricting evaluation to large-cap equities limits generalisation to asset classes with different microstructures. Furthermore, while deterministic inference reduces instability, the system remains sensitive to retrieval quality, prompt specification, and inherent LLM training biases or data leakage.

Future work will explore three directions: incorporating term-structure and volatility surface features to improve regime sensitivity; integrating risk-sensitive objectives to stabilise extreme-event performance [11]; and generalising the architecture to structured multi-agent settings where specialised controllers interact via explicit coordination protocols [10].

6 Conclusion

Financial decision-making under market regime shifts often suffers from instability and increased downside risk. While recent approaches leverage large language models or end-to-end learning, it remains unclear whether their success stems from raw model scale or structural design. This raises a key question: can

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39

conditioning decisions on retrieved historical precedents systematically improve system robustness?

To address this, we introduced *MultiHedge*, a hybrid architecture that combines structured decision rules with LLM-based reasoning guided by retrieved historical examples. Our results show consistent improvements in performance and substantial reductions in downside risk compared to both rule-based and learning-based baselines. Ablation studies indicate that memory and system design play a larger role than model size in driving these gains. Overall, this paper contributes to the field of robust decision systems by showing that combining memory with modular decision structures can improve performance under changing conditions.

References

1. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Communications **7**(1), 39–59 (1994)
2. Bańka, F., Chudziak, J.A.: Options pricing platform with neural networks, llms and reinforcement learning. In: Recent Challenges in Intelligent Information and Database Systems. pp. 202–216. Springer Nature Singapore (2025)
3. Bańka, F., Chudziak, J.A.: Deltahedge: A multi-agent framework for portfolio options optimization. In: PACIS 2025 Proceedings. No. 25 (2025)
4. Buehler, H., Gonon, L., Teichmann, J., Wood, B.: Deep hedging. Quantitative Finance **19**(8), 1271–1291 (2019)
5. Guo, T., et al.: Large language model based multi-agents: A survey of progress and challenges (2024)
6. Kirtac, K., Germano, G.: Sentiment trading with large language models. Finance Research Letters **62**, 105227 (Apr 2024)
7. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., et al.: Retrieval-augmented generation for knowledge-intensive nlp tasks. In: Advances in Neural Information Processing Systems (NeurIPS) (2020)
8. Liberzon, D.: Switching in Systems and Control. Birkhäuser (2003)
9. Liu, X.Y., Xiong, Z., Zhong, S., Yang, H., Walid, A.: Practical deep reinforcement learning approach for stock trading (2022)
10. Macal, C.M., North, M.J.: Tutorial on agent-based modelling and simulation. Journal of Simulation **4**(3), 151–162 (2010)
11. Malekzadeh, P., Poulos, Z., Chen, J., Wang, Z., Plataniotis, K.N.: Ex-drl: Hedging against heavy losses with extreme distributional reinforcement learning (2024)
12. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley (1994)
13. Shu, Y., Yu, C., Mulvey, J.M.: Dynamic asset allocation with asset-specific regime forecasts (2024)
14. Szydlowski, K.L., Chudziak, J.A.: Toward predictive stock trading with hidformer integrated into reinforcement learning strategy. In: Proc The 36th International Conference on Tools for Artificial Intelligence (ICTAI 2024) (2024)
15. Xiao, M., Jiang, Z., Qian, L., Chen, Z., He, Y., Xu, Y., Jiang, Y., Li, D., Weng, R.L., Peng, M., Huang, J., Ananiadou, S., Xie, Q.: Retrieval-augmented large language models for financial time series forecasting (2025)
16. Zhang, W., et al.: A multimodal foundation agent for financial trading: Tool-augmented, diversified, and generalist (2024)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29924-6_39