Hi-DARTS: Hierarchical Dynamically Adapting Reinforcement Trading System

HOON SAGONG
Dept. of Computer Engineering
Hongik University
Seoul, South Korea
clap518@mail.hongik.ac.kr

HEESU KIM
Dept. of Applied Data Science
Sungkyunkwan University
Seoul, South Korea
k.heesu@skku.edu

HANBEEN HONG
Dept. of Economics
Hankuk University of Foreign Studies
Seoul, South Korea
hhb0618@hufs.ac.kr

***Abstract*—Conventional autonomous trading systems struggle to balance computational efficiency and market responsiveness due to their fixed operating frequency. We propose Hi-DARTS, a hierarchical multi-agent reinforcement learning framework that addresses this trade-off. Hi-DARTS utilizes a meta-agent to analyze market volatility and dynamically activate specialized Time Frame Agents for high-frequency or low-frequency trading as needed. During backtesting on the AAPL stock from January 2024 to May 2025, Hi-DARTS yielded a cumulative return of 25.17% with a Sharpe ratio of 0.75. This performance surpasses standard benchmarks, including a passive buy-and-hold strategy on AAPL (12.19% return) and the S&P 500 ETF (SPY) (20.01% return). Our work demonstrates that dynamic hierarchical agents can achieve superior risk adjusted returns while maintaining high computational efficiency.**

***Index Terms*—Algorithmic Trading, Reinforcement Learning, Proximal Policy Optimization, Multi-Agent Systems, Adaptive Framework**

I. Introduction

The proliferation of high-frequency trading in financial markets has made automated trading systems essential to achieve competitive yields. However, these systems struggle with balancing computational efficiency with market responsiveness. Most frameworks active today operate at a fixed frequency, consuming computational resources in a consistent manner. This is a static approach that is either computationally wasteful during calm market periods or too slow to capitalize on opportunities during periods of high volatility. This trade-off between cost and performance creates a demand for adaptive solutions. Recent advances in deep reinforcement learning have shown promising results in high-performance decision-making problems [1]. Multi-agent reinforcement learning frameworks have been shown to perform effectively for complex tasks [2]–[4]. To address this challenge, we introduce Hi-DARTS, a novel Hierarchical Dynamically Adapting Reinforcement Trading System. Our framework utilizes a two-layer structure of Proximal Policy Optimization (PPO) agents. The top-level agent, the Time Frame Allocator, analyzes real-time market data and employs one of several specialized Time Frame Agents, each trained to operate optimally at a different frequency (e.g., 1 hour, 10 minute, or 1 minute). This hierarchical design allows the system to conserve resources in stable markets and remain capable of reacting to fast changes.

The main contributions of this paper are threefold:

- We propose a novel hierarchical multi-agent architecture for algorithmic trading that dynamically adapts its operational frequency to market conditions.
- We introduce a dynamic allocation mechanism where a meta-agent effectively learns to select the optimal specialized trading agent based on volatility.
- We provide empirical validation of our framework on real-world stock data, demonstrating that Hi-DARTS achieves superior risk adjusted returns compared to static benchmarks.

II. Proposed Hierarchical Framework

The proposed framework consists of a two-layered hierarchical structure, as illustrated in Fig. 1. This top-down design is a modular approach that has numerous benefits in this particular system, such as the specialization of each task, Time Frame Agent selection, and decision-making. This specialization allows the system to perform better than the traditional layerless trading system [5]. Then, by using a risk-based reward system with each layer, the system has the ability to evaluate its own performance and further strengthen its capabilities. With the help of propagated rewards, the Time Frame Allocator can select the appropriate Time Frame Agents.

A. Stock Layer (Time Frame Allocator)

This layer is the adaptive mechanism and functions as a Time Frame Agents Allocator. It continuously analyzes the selected stock's recent market data with the rewards produced from the Time Frame Agents to assess the current market state of the stock. Based on this assessment, it determines which of the Time Frame Agents is most suitable for the current conditions and should be designated. For example, in a highly volatile market, it would activate a high-frequency agent (e.g., 1 minute agent), whereas in a stable market, it would initiate a low-frequency agent (e.g., 1 hour agent) to conserve resources.

B. Time Frame Agents (Market Responders)

At the lowest level, the Time Frame Agents interact directly with the raw stock market data and come up with the judgment to buy, sell, or hold. These agents are also tasked with making

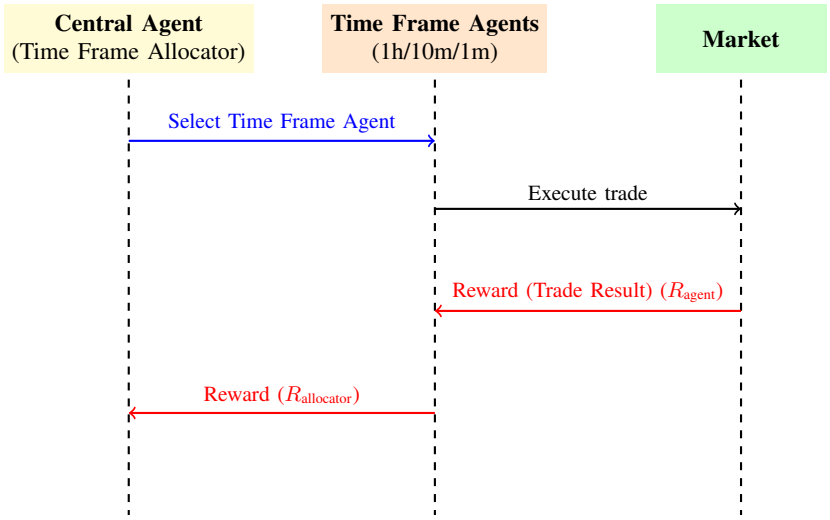

Fig. 1. Reward flow in the adaptive framework: The trade outcome from the Market provides a direct reward (R_{agent}) to the selected Time Frame Agent for its own learning. This outcome is then used to generate a subsequent reward ($R_{\text{allocator}}$) for the Central Agent, allowing it to learn the optimal allocation strategy.

the transaction itself, acting semi-autonomously since they are directly managed by the upper layer. The agents are trained independently, using only the data with the same time resolution. By limiting the data these agents can view, they are able to better analyze the given data since they are more specialized to deal with data with such time resolution. When deployed, they are each fed with relevant data, and each of them comes up with their own judgment of the situation by reacting to the real-time data. Consequently, after making such actions, rewards are produced from the outcome of their actions.

1) Technical Indicators and Variables: Table I contains the features selected for our models. These technical indicators and portfolio statistics serve as input variables for each Time Frame Agent to inform trading decisions at their respective operational frequency. The selected features are as follows:

- RSI 14: The 14-period Relative Strength Index, a momentum oscillator that measures the speed and change of price movements, helping to identify overbought or oversold conditions.
- MACD Hist: The Moving Average Convergence Divergence Histogram, which captures the difference between the MACD line and its signal line and provides insight into the strength and direction of momentum.
- CCI20: The 20-period Commodity Channel Index is used to identify cyclical trends in security prices and detect deviations from typical price ranges.
- BB pband 20: The 20-period Bollinger Bands %B, describing the position of the latest closing price with respect to the Bollinger Bands and indicates potential breakouts or mean reversion opportunities.
- Volume: The total number of shares traded over the specified period, reflecting market activity and liquidity.
- Cash Ratio: The proportion of the agent's portfolio held in cash, calculated as the cash balance divided by the total portfolio value. This ratio provides information on available liquidity and the risk management posture.
- Stock Ratio: The proportion of the agent's portfolio invested in equities, computed as the stock value divided by the total portfolio value, which indicates market exposure.
- Unrealized Profit Ratio: The percentage profit or loss of currently held equity positions, which allows the agent to assess ongoing trade performance and adjust actions accordingly.

TABLE I
SELECTED FEATURES AND DESCRIPTIONS

Features	Descriptions
RSI_14	14-period Relative Strength Index
MACD_Hist	Moving Average Convergence Divergence histogram
CCI20	20-period Commodity Channel Index
BB_pband_20	20-period Bollinger Bands %B
Volume	Total number of shares traded over a period
Cash_Ratio	Cash balance divided by total portfolio value
Stock_Ratio	Stock value divided by total portfolio value
Unrealized_Profit_Ratio	The profit or loss in percentage of the equity held

These indicators collectively enable the Time Frame Agents to capture momentum, volatility, trend, market activity, and portfolio health, supporting robust decision-making across different market regimes.

III. IMPLEMENTATION AND VALIDATION

We implemented our proposed framework using Proximal Policy Optimization (PPO) as the base algorithm for our

agents. PPO was chosen because it has shown superior performance over policy gradient methods [6]. All experiments were conducted on a single GPU (Tesla V100). All data used to train, validate, and test came from the Wharton Research Data Services (WRDS). To validate this implementation, we designed benchmarks that demonstrate the ability of the allocator to make the optimal decision when given the data. Consequently, the Time Frame Agents' (1 minute, 10 minute, 1 hour) and Central Agent's (Time Frame Allocator) architecture consists of a two hidden layer MLP for policy and value networks. Key hyperparameters, such as learning rate, state window size, and total timesteps, were individually tuned for each agent's respective timeframe to optimize its performance. All hyperparameters for each agent are depicted in Table II.

A. Experimental Setup

In the experimental setup for this proof-of-concept implementation, our system was trained and evaluated on Apple Inc. (AAPL) stock data. Three different Time Frame Agents were trained on 1 minute, 10 minute, and 1 hour interval data. The dataset was divided into three parts: training, validation, and testing. The training set consists of data from January 2, 2013, to December 31, 2022, the validation set from January 2, 2023, to December 31, 2023, and the test set from January 2, 2024, to May 2, 2025. Each Time Frame Agent was trained for 200,000 timesteps in total, with an initial cash budget of $10,000. The Stock Layer (Time Frame Allocator) was then trained on the 1 minute data from the training set with pretrained Time Frame Agents attached.

B. Performance Metrics

To quantitatively validate our framework, we tested its performance against two standard benchmarks: a Straightforward Buy & Hold strategy for the AAPL stock, and the S&P500 Index ETF. Our evaluation focused on the following metrics:

- Cumulative Return: The total gain over a given period of time.
- Sharpe Ratio: Risk-adjusted performance compared to a risk-free investment.
- Maximum Drawdown (MDD): Measure of the largest drop in the portfolio's value within the entire evaluation period.

The Sharpe Ratio measures the performance of an investment by its risk-adjusted return as described in [7]. Maximum Drawdown, a standard risk metric in quantitative finance, measures the peak-to-trough decline of the investment for a given time period, as described in [8].

TABLE II
DETAILED HYPERPARAMETERS FOR ALL AGENTS USED IN THE HI-DARTS FRAMEWORK.

Hyperparameter	Allocator	1m Agent	10m Agent	1h Agent
PPO Algorithm Parameters				
Total Timesteps	300,000	500,000	200,000	150,000
Learning Rate	3e-4	5e-5	1e-4	1e-4
N_Steps	2048	4096	2048	1024
Batch Size	256	128	128	64
N_Epochs	10	10	10	10
Gamma	0.99	0.99	0.99	0.99
GAE Lambda	0.95	0.95	0.95	0.95
Clip Range	0.2	0.2	0.2	0.2
Entropy Coef.	0.005	0.01	0.03	0.01
Environment & State Parameters				
Window Size	N/A	240	120	80
Initial Cash	$10,000	$10,000	$10,000	$10,000
Network Architecture				
MLP Layers	2x64	2x256	2x64	2x256

C. Reward Function

Hi-DARTS uses the following reward functions for the Time Frame Agents and the Time Frame Allocator. The reward function is designed to reflect how profitable an executed trade is. Equation (1) is defined as a hyperbolic tangent transformation of the normalized difference between the selling price and the average buying price of the stock,

$$R_{\text{agent}} = \tanh\left(5 \times \frac{P_{\text{sell}} - P_{\text{buy,avg}}}{P_{\text{buy,avg}}}\right) \tag{1}$$

where P_{sell} is the selling price and $P_{buy,avg}$ is the average purchase price of the stock held. This formula ensures that the reward remains bounded between -1 and 1, providing stable gradients for training while emphasizing the relative profit made by each transaction.

The Time Frame Allocator, which dynamically selects among the specialized agents, receives feedback based on the overall portfolio value changes. Its reward $R_{allocator}$ is defined in (2) as the natural logarithm of the ratio between the current portfolio value $V_{current}$ and the previous portfolio value $V_{previous}$:

$$R_{\text{allocator}} = \ln\left(\frac{V_{\text{current}}}{V_{\text{previous}}}\right) \tag{2}$$

This logarithmic return metric captures the agent's ability to allocate trading decisions in a manner that maximizes portfolio growth while implicitly penalizing losses, facilitating the learning of an optimal allocation strategy that adapts to market dynamics.

By adopting these reward functions, Hi-DARTS effectively balances the micro-level trade execution performance of individual Time Frame Agents with the macro-level portfolio management objectives of the allocator, leading to improved risk-adjusted returns and adaptive trading behavior. In order to have an immediate response, the 1 minute agent is selected by default at the start of the market. This helps the system react spontaneously and produce rewards as soon as the market opens. Also, the system is designed to liquidate all stock holdings by market close, aiming to optimize outcomes and constructively enhance agent performance.

D. Stock Layer Validation

Historical data from January 2, 2013, to December 31, 2022, was used to train the Time Frame Agents. This period contains both volatile and stable periods that taught the models to react to both types of markets. The allocator decide upon which Time Frame Agent to allocate based on previous data and

reward feedback. The system was given data from January 2, 2021, to December 31, 2023, for the validation of the entire model. The allocator successfully opted for shorter frequency agents during periods of turbulent price movements and switched to longer-frequency agents when the market was calm. This result confirms that our adaptive allocation mechanism produces the intended outcome, laying a solid foundation for the full system.

To analyze the designation of the agents between 1 minute, 10 minute, and 1 hour agents, we have analyzed the selection of agents monthly, daily, and hourly. To examine the allocation, the entire trade has been divided into quartiles, from the lowest volatility to the highest, as shown in Fig. 3. For this analysis, daily volatility was calculated as the standard deviation of price returns multiplied by 100.

Starting with the monthly report of agent allocation, the lowest quartile segment has shown the 0.18% average pick rate for the 1 minute agent, while the average pick rate for the 1 hour agent was 11.03%. Then the second segment exhibited the 0.25% utilization for the 1 minute agent, and 11.57% usage of the 1 hour agent. For the third quartile, the 1 minute agent was selected 0.40% and the 1 hour agent was selected 10.43%. Lastly, in the most volatile quartile, the pick rate for 1 minute has increased to 0.71%; meanwhile, 1 hour rate was decreased to 7.90%.

Subsequently, the same trend continued in the daily average volatility, too. The first quartile had not employed the 1 minute agent at all, while the system used the 1 hour agent 13.74% of the time. In a similar manner, the second quartile also had not selected 1 minute agent for trading, whereas it had chosen 1 hour agent 12.04%. Next, the third quartile deployed 1 minute agent 0.05%, whilst 1 hour agent was designated 9.34%. Lastly, the most volatile quartile had 1 minute agent 1.40% of the time, though it used 1 hour agent 7.49%.

For the more granular examination, we have analyzed the hourly designation of the agents. Similarly to the analysis above, the first quartile did not employ 1 minute agent at all, although it had employed 1 hour agent 18.54%. Likewise, the second quartile had zero usage of 1 minute agent, but the system employed 1 hour agent 10.99%. Correspondingly, the third quartile also did not use 1 minute agent; However, it deployed the 1 hour agent 6.95%. The final quartile had 1.45% of 1 minute agent and 6.19% of 1 hour agent.

With every month, day, hour agent selection, it clearly shows the trend; for the high volatility, the system opted for more 1 minute agent, subsequently fewer 1 hour agent. This has shown meaningful results from our stock layer in terms of the volatility it faces in the stock market.

E. Result

In this paper, we have introduced a hierarchical framework for an adaptive stock trading system. By dynamically allocating decision-making frequency based on ever-changing market conditions, our approach promises to balance computational efficiency with immediate market responsiveness. Our preliminary results validate the core component, the Time Frame Allocator, which successfully distinguishes different market states. Fees or taxes are intentionally excluded from our evaluation in order to generalize the measured performance across different environments, where transaction fees may vary. However, even with a standard transaction fee of 1 cent per sell, the final portfolio value's difference remains below 1%.

Subsequently, the historical data from January 2, 2024, to May 2, 2025, was used to evaluate the system. The system was initially assigned with $10,000.00, and the final portfolio was worth $12,517.04, which produced a 25.17% cumulative return (Table III). In comparison to the Buy & Hold, which represents $10,000 worth of stock purchased and held without being sold, the final valuation of that portfolio was $11,218.51 during the same period, generating a 12.19% rate of return (Table III). In addition, the S&P rate of return during the same period was 20.01%, making the S&P500 portfolio worth $12,001.43 (Table III).

IV. Conclusion and Future Work

Conventional automated trading systems are limited to a fixed computing frequency, regardless of the market conditions. They fail to adapt to a dynamic market. Consequently, if the performance of the automatic trading system is excessively set, it will waste resources in the calm markets. However, if minimized, it would not have enough responsiveness for the chaotic stock market.

Therefore, multi-agent frameworks can be more efficient in real-life situations. Hi-DARTS can save computational resources in the dynamic stock market. The product of this study validates the substantial correlation between the market's status and the agents' allocation.

As demonstrated in the validation above, this system minimizes the waste of computational power compared to the traditional system. As the volatility level increases, the system adjusts the deployment ratio between the agents. In contrast, according to Fig. 3, at the first and second quartile, Hi-DARTS utilizes 0% of 1 minute agent. This illustrates that at the low volatility levels, when the immediate response is not required, the system acts passively to save processing power. This implication is evidently exhibited by the Fig. 3 bar graphs, which convey Hi-DARTS making agent transitions in different volatility levels.

As depicted from Fig. 3, when the market moved from the least volatile quartile to the most fluctuating quartile, the allocation of 1 minute agents was increased from 0% to 1.4%, while the 1 hour agent deployment decreased from 13.7% to 7.5%. When the market was volatile, the system made 1 minute agent more prominent to immediately respond to the fluctuations, meanwhile reducing the use of 1 hour agent to minimize the response time.

Our future work will focus on completing the full suite of Time Frame Agents using PPO. We will also conduct a comprehensive performance evaluation by comparing our adaptive system's profitability and computational cost against the fixed-frequency automatic trading systems.

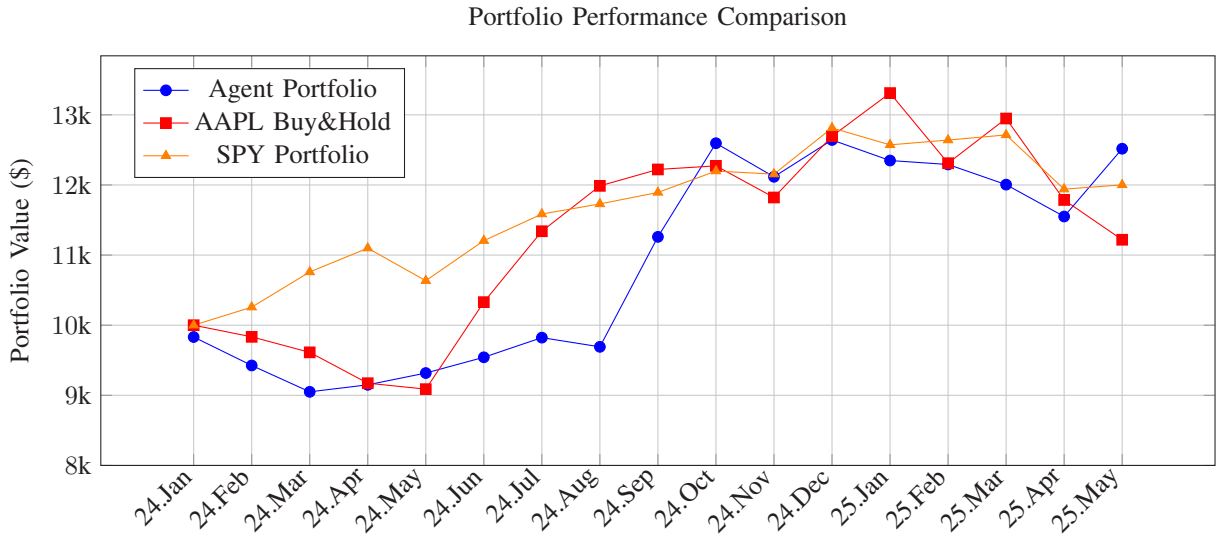

Fig. 2. Comparison of the performance of Hi-DARTS and passive buy-and-hold strategies for AAPL and SPY during the test set time period. The test period is defined to be from January 2, 2024, to May 2, 2025. Hi-DARTS achieved a cumulative return of 25.17% compared to the 12.19% and 20.01% of AAPL and SPY, respectively.

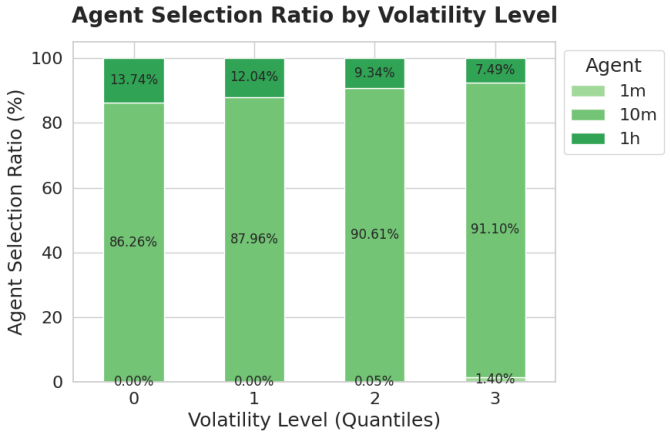

Fig. 3. Agent selection ratios are divided by daily volatility. The daily volatility of the market was calculated and then sorted into four equal quartiles, from lowest (0) to highest (3). This chart shows the average agent selection ratio for all days within each volatility quartile, confirming that the system activates high-frequency agents during volatile periods and low-frequency agents during stable periods.

TABLE III
PERFORMANCE COMPARISON WITH BENCHMARKS AND RELATED WORKS

Model	Test Period	Return (%)	Sharpe Ratio	MDD (%)
Hi-DARTS	**2024.01–2025.05**	**25.17**	**0.75**	**-25.49**
1 Minute Agent	2024.01–2025.05	-8.06	0.04	-25.78
10 Minute Agent	2024.01–2025.05	10.16	0.35	-18.68
1 Hour Agent	2024.01–2025.05	-1.38	0.05	-21.67
Buy & Hold (AAPL)	2024.01–2025.05	12.19	0.35	-33.36
S&P 500 (SPY)	2024.01–2025.05	20.01	0.58	-18.76
PPO (DJIA) (FinRL [9])	2019.01.01–2020.09.23	18.53	0.48	-37.01

We plan to implement the Central Layer (Stock Selector) on top of the hierarchical model. The top layer of the hierarchy will be the Central Layer. Its primary responsibility is to analyze the stocks, depending on the existing data set. This layer will provide a specific stock to the lower layers. After finalizing all three layers of Hi-DARTS, we will apply other stocks to the system. The ultimate goal is to have the model seamlessly adapt to new assets without requiring additional training, due to its adaptive reward system.

Subsequent to finalizing the full suite, we plan to expand the number of Time Frame Agents to effectively cover the volatility of the perpetually shifting stock market. Then we plan to make an override mechanism that can overrule the initial designation of a low-frequency agent to a high-frequency agent that is more reactive. This can assist the system to remain autonomous, even when the market faces externalities that abruptly alter the market, such as major geopolitical events or shocks that shake the stock market.

Consequently, our objective is to create a fully autonomous stock trading system that can save computational power while remaining responsive and effective.

REFERENCES

[1] A. Mosavi, P. Ghamisi, Y. Faghan, and P. Duan, "Comprehensive review of deep reinforcement learning methods and applications in economics," *Mathematics*, vol. 8, no. 10, p. 1640, Oct. 2020.

[2] R. Lowe, Y. Wu, A. Tamar, J. Ba, P. Abbeel, and I. Sutskever, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 6379–6390.

[3] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *Proc. Int. Conf. Auton. Agents Multiagent Syst. (AAMAS)*, 2017, pp. 66–74.
[4] T. Spooner, J. Fearnley, R. Savani, and A. Kouvaris, "Market-based multi-agent reinforcement learning for intelligent energy trading," in *Proc. 17th Int. Conf. Auton. Agents Multiagent Syst. (AAMAS)*, 2018, pp. 2146–2148.
[5] C. Ye, X. Liu, K. Wang, and B. Liu, "Multi-scale asset management: A hierarchical reinforcement learning approach," in *Proc. 29th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2020, pp. 4627–4633.
[6] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.
[7] W. F. Sharpe, "The Sharpe ratio," *J. Portfolio Manag.*, vol. 21, no. 1, pp. 49–58, 1994.
[8] M. J. Magdon-Ismail and A. F. Atiya, "Maximum drawdown," *Risk*, vol. 17, no. 10, pp. 99–102, 2004.
[9] X. Liu, H. Yang, Q. Wang, and J. Liu, "FinRL: A deep reinforcement learning library for automated stock trading in quantitative finance," in *Proc. 2nd ACM Int. Conf. AI Finance*, 2021, pp. 1–8.