Investor Preference-Aware Portfolio Optimization with Deep Reinforcement Learning

Zhenglong Li, Vincent Tam, Kwan L. Yeung
Department of Electrical and Electronic Engineering, The University of Hong Kong
E-mails: lzlong@connect.hku.hk, {vtam, kyeung}@eee.hku.hk

***Abstract*—Recently, deep reinforcement learning (RL) approaches have been adopted to optimize financial portfolios with the objective of maximizing total profits while reducing potential risks by spreading investment capital across a variety of assets. Despite achieving great advances in the trade-off between profits and risks, the existing deep RL-based frameworks rarely consider practical trading constraints when making decisions in real-world financial markets, which cannot fulfill the customized requirements of specific users and may violate market regulations. Accordingly, a Multi-Agent and Self-Adaptive trading framework for Constrained portfolio optimization, namely the MASAC, is proposed in which the deep RL-based agent dynamically explores profit-maximization policies while the heuristic-based agent conducts min-conflict search to ensure the generated trading strategies satisfying all concerned constraints. Through sharing knowledge within the trading system, the agents of the proposed framework cooperatively produce new portfolios that can maximize overall profits while satisfying all investor requirements throughout the trading period. The experimental results reveal the advantages of the MASAC framework against many state-of-the-art approaches in investment performance and trading constraint satisfaction on the challenging data sets of real-world markets.**

***Index Terms*—Trading System, Financial Portfolio Optimization, Risk Management, Deep Reinforcement Learning**

I. Introduction

Portfolio optimization, as one of the active research topics in the field of computational finance, aims at maximizing the overall returns and reducing investment risks through dynamically adjusting the investment weights of assets in a portfolio to adapt to the ever-changing financial market over the trading period. However, except for balancing the returns and risks, in fact that there are many trading constraints to be considered in the real-world markets to fulfil the requirements of government regulations and investor preferences. Therefore, making the trade-off between investment returns, potential risks, and trading constraints can be a very challenging task in Constrained Portfolio Optimization Problems (CPOP).

Conventionally, fund managers make trading decisions based on their intuition and past experiences, yet it may lead to a biased strategy due to subjective judgments. Following modern portfolio theory [1], the CPOP is formulated as a mathematical programming problem and optimized by dynamic programming solvers [2]. Nevertheless, dynamic programming solvers restrict the form of constraints and fail to forecast the future trends of asset prices from historical data. Recently, deep or reinforcement learning (DL/RL) has achieved great success in many practical financial applications including order execution [3], high-frequency trading [4], and pair trading [5], among which many studies [6], [7] have attempted to utilize DL/RL-based approaches to learn the price movement of assets as well as the correlations between assets in a portfolio under the highly volatile financial markets. Yet in many cases, even if those DL/RL-based approaches are good at learning profitable trading strategies, they cannot effectively handle trading constraints due to the increased difficulty of policy training when optimizing different objectives in the same reward function, especially together with multiple types of constraints or the non-differentiable constraints, thus those approaches cannot be customized for different investors and are inapplicable in real-world trading systems.

To overcome the above pitfalls, a novel **M**ulti-**A**gent and **S**elf-**A**daptive trading framework for **C**onstrained portfolio optimization, namely the MASAC, is proposed in this work in which a cooperative agent-based scheme is adopted to carefully balance the trade-off between the overall returns, potential short-term risks, and the satisfaction of multiple trading constraints during the whole trading period. The deep RL-based agent of the MASAC framework captures the underlying trend patterns of each asset and the correlations between involved assets from historical price data, dynamically generating new portfolio weights with high long-term profit expectations for adapting to the volatile financial markets. Based on the priori knowledge provided by the RL-based agent, the cooperating heuristic-based agent namely the EGENET+ exploits the neighboring area of the generated portfolios, trying to find a minimum change of the generated portfolios such that the trading decision can satisfy all the assigned constraints while maintaining the expectations of high long-term profits. Besides, the constraint conflict information in the heuristic-based agent is returned to the deep RL-based agent to guide solving difficult constraints. The main contributions of the proposed framework are summarized as follows.

1) Compared with the existing approaches, the MASAC framework achieves a good balance between investment performance and practical trading constraints in real-world trading by the cooperation between two agents.
2) There is no limit to the types of constraints and no need to retrain the RL models when changing trading requirements. Therefore, the MASAC can customize trading strategies for investors with different requirements.
3) As a flexible and practical trading framework, the user

can conveniently apply the MASAC to various financial products in different financial markets around the world.

II. Related Works

Traditional Portfolio Optimization. Conventionally, many investors adopted trend-tracking strategies [8] to decide buying or selling assets. The follow-the-winner strategies [9] prefer to invest the well-performing assets in the past period while the follow-the-loser strategies [10], [11] believe that the worse-performing assets will rise back to the normal soon. However, the financial market is ever-changing, thus those trading strategies may not be adaptive to different market states.

Solver-based Portfolio Optimization. As aforementioned, through formulating the CPOP as a dynamic programming problem, there are many powerful solvers [12]–[14] that can be applied to solve CPOP. Yet some trading constraints involve special items like the L1-norm, or the constraints cannot be even defined in formulas. This may not be dealt with by traditional solvers. Moreover, those solvers lack the ability to estimate the future price trends of assets from historical data.

Deep Reinforcement Learning-based Portfolio Optimization. Deep reinforcement learning, as one of the widely used portfolio optimization techniques in recent years, demonstrates great potential for capturing both the future trend patterns of asset prices and the correlations between assets. The convolution-based trading framework [15] tries to optimize portfolio weights and reduce turnover rates while [7], [16] investigate the effects of investment risks and transaction costs with a recurrent-convolution framework. Followed by the success of attention-based architectures, [6], [17] present two transformer-based models to optimize asset allocation by the relation attention and sequential attention mechanisms. With the rapid development of multi-agent RL systems, there are other studies [18], [19] investigating the potential use of multiple intelligent agents to cooperatively work in the field of portfolio management, among which [18] proposes different agents to maximize returns and manage short-term risks. However, the previous RL-based approaches do not take into account practical trading constraints, or only consider one constraint such as investment risks or margin requirements.

III. Methodology

A. Problem Formulation

The CPOP aims to maximize overall returns while satisfying all given trading constraints during the trading period T.

$$\begin{aligned} &\max \sum_{t=1}^{T} \log\left(\mathbf{r}_t^{\top}\mathbf{a}_{t-1}+1\right) \\ \text{s.t.}\quad &\mathrm{CSTR}_{j,t}, \forall j \in J, \forall t \in T, \end{aligned} \tag{1}$$

where $\mathbf{a}_{t-1} \in \mathbb{R}^N$ is the weight vector of assets at time $t-1$, $\mathbf{r}_t \in \mathbb{R}^N$ is the daily return rates of assets in a portfolio, and $\mathrm{CSTR}_{j,t}$ denotes the j^{th} constraint at t. As described in [20], seven government regulations and investor requirements are presented by mathematical formulas and added to the trading constraints in this paper, including 1) Short-term risk constraint based on the variance of Markowitz model [1] for reducing huge losses in a short time; 2) Holding constraint that manages the range of investment weights of specific stocks in terms of investor preference; 3) Cardinality constant that diversifies investment risks by restricting the minimum and maximum number of assets in a portfolio at t; 4) Turnover constraint that keeps the consistency of trading behaviors and subject to regulatory restrictions through managing the total trading quantities of assets in each transaction; 5) Class constraint that manages the investment range of a subset of assets recommended by investors; 6) Industry constraint that restricts the total weights of assets in the same industry; 7) Small capitalization constraint that limits the investment proportion in the stocks with a relatively small market capitalization. The types of constraints include linear constraints, second-order cone constraints, and L1-norm constraints.

B. The Overview of the Trading System

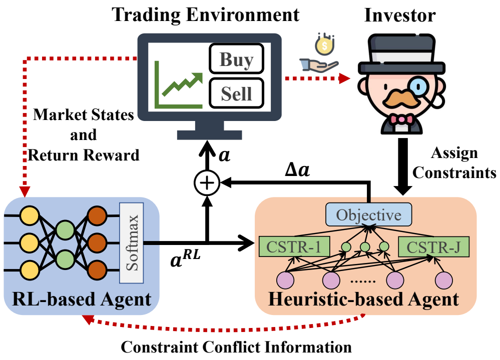

Fig. 1. The System Architecture of the Proposed MASAC Framework

To overcome the pitfall of the existing portfolio optimization approaches on balancing the expectations of long-term returns and trading constraints satisfaction throughout the trading period, a **M**ulti-**A**gent and **S**elf-**A**daptive framework for **C**onstrained portfolio optimization namely the MASAC is proposed in this work. Through the clear division of works to optimize different objectives and the close cooperation to share valuable information, the two agents, namely the deep RL-based agent and heuristic-based agent, construct a new agent-based RL scheme to dynamically generate trading signals to adapt to the ever-changing financial markets.

The overall system architecture of the MASAC framework is shown in Fig. 1. Unlike the constraint conflict as a penalty item is integrated into the loss function in most of the DL or RL-based portfolio management approaches to simultaneously optimize investment returns and trading constraints, the MASAC framework as a divide-and-conquer approach separates the two optimization tasks to different agents in which the deep RL-based agent focuses on maximizing the overall returns while the heuristic-based agent fine-tunes the trading strategies for satisfying the trading constraints assigned by investors. First of all, the deep RL-based agent learns the price movement of assets and the correlations between assets by capturing the valuable patterns of historical price data, continuously producing portfolio weights $\mathbf{a}^{RL}$ to ultimately achieve higher long-term profits. Then, the heuristic-based agent receives and transforms investor instructions to trading constraints. After that, according to the priori knowledge $\mathbf{a}^{RL}$, the heuristic-based agent executes the min-conflict local search around the promising space recommended by the RL-based agent such that the newly revised portfolio $\mathbf{a}$ with the minimum changes $\Delta\mathbf{a}$ can satisfy all constraints while maintaining the relatively high long-term returns expected by the RL-based agent as possible. With the utilization of constraint conflict information provided by the heuristic-based agent, it is worth noting that the cooperation mechanism of the MASAC can further guide the RL-based agent to adaptively explore more promising regions for avoiding local minima and reach a new balance between portfolio performance and difficult constraint satisfaction. More specifically, the heuristic-based agent indicates the constraints which may be difficult to solve at the current market states by itself, and then the RL-based agent can help explore the possible solutions to satisfy such constraints by assigning extra penalties and different sampling probabilities of training samples.

C. The EGENET+ Solver for Constraint Satisfaction

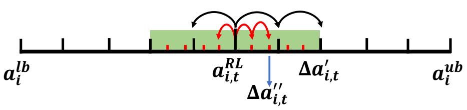

Fig. 2. An Illustration of the Enhanced Search Scheme

Originally, the EGENET [21] was designed for continuous constrained problems like graph colouring optimization by using min-conflict heuristic search. In this paper, an enhanced version of EGENET, namely the EGENET+, working as the heuristic-based agent in the MASAC framework, is proposed to collaborate with the RL-based agent for further enhancing the ability to solve different types of trading constraints. To reduce the modifications of the portfolio $\mathbf{a}_t^{RL}$ for maintaining the expectations of long-term returns by the RL-based agent while satisfying all trading requirements, the EGENET+ tries to minimize the compensated action $\Delta\mathbf{a}_t$ such that all constraints can be satisfied. As illustrated in Fig. 2, the $a_{i,t}^{RL}$ is set to the start point of the search. The EGENET+ alternately conducts the forward and backward search (the black arrows) until finding a feasible solution $\Delta a'_{i,t}$. Subsequently, the search step size and the search boundary of variable i are shrunk to the shadow region in green as the area outside of the green-shaded area does not exist a solution that is less than $\Delta a'_{i,t}$. After conducting a more fine-grained search (the red arrows), the EGENET+ will recognize a smaller compensated action $\Delta a''_{i,t}$ that can fulfill all constraints. Finally, the best $\Delta\mathbf{a}_t$ is returned to produce the final portfolio $\mathbf{a}_t$.

D. The Guided Learning for Trading Policy Optimization

In general, portfolio optimization can be formulated as a partially observable Markov decision process in multi-period trading and optimized by deep reinforcement learning with its remarkable ability on sequential decision-making problems. The observable states include the open, high, low, and close price of assets in a portfolio. The main objective of the RL-based agent is to maximize the overall returns $L_r(\theta_1) = \frac{1}{T}\sum_{t=1}^{T}\log\left(\mathbf{r}_t^\top \mathbf{a}_{t-1} + 1\right)$ and explore the solutions to handle difficult constraints $L_c(\theta_2) = \frac{1}{T}\sum_{t=1}^{T}\frac{J_{sat,t}}{J_t}$. Thus, the customized reward function of the RL-based agent is depicted as $L(\theta_1, \theta_2) = \lambda_1 L_r(\theta_1) + \lambda_2 L_c(\theta_2)$, where λ_1 and λ_2 are the learning weights of reward items to select conservative or aggressive strategies, $J_{sat,t}$ is the number of satisfied constraints and J_t is the total number of constraints at t. Besides, the sampling probability of the training sample k is updated as $\Pr_k = \frac{J_k - J_{sat,k}}{\sum_{m=1}^{M} J_m - J_{sat,m}}$, where M is the total number of training samples. By adaptively adjusting $\Pr_k$, the RL-based agent of the MASAC is guided to focus on the samples with more violated trading constraints, which helps to escape from local minima and attains a better balance between trading performance and constraint satisfaction.

IV. Experiments

A. Experimental Settings

Two real-world data sets of S&P 500 and Dow Jones Industrial Average (DJIA) indexes from January 2014 to March 2024 are collected to compare the performance of the MASAC and other state-of-the-art approaches, among which the first five-year data for training, subsequent two-year data for validating, and the last three-year data for testing. The top 10 stocks of a market index are selected to construct the portfolio in terms of their market capitalization. All the reported results are averaged over 5 runs for reducing randomness impacts. To compare the performance of the MASAC, 11 state-of-the-art methods based on different optimization techniques are selected, including 1) Three traditional approaches: EG [9], PAMR [10], and RMR [11]; 2) Three solver-based methods: XPRESS [14], GUROBI [13], and CPLEX [12]; and 3) Five deep RL-based frameworks: DPM [15], PPN [16], RAT [17], DT [7], and MASA [18]. Besides, four widely used metrics in previous works [15], [18] including Annual Return (AR), Maximum Drawdown (MDD), Sharpe Ratio (SR), and Volatility (Vol) are adopted to evaluate investment performance. Also, to evaluate the comparative methods in handling multiple constraints, two popular metrics in the constraint satisfaction area are selected. The Feasible Day (FD) is the number of trading days satisfying all constraints in the trading period T while the Constraint Satisfaction (CS) rate is the average number of satisfied constraints per day.

B. Performance Analysis

Table I compares both constraint satisfaction and investment performance of the MASAC framework against the state-of-the-art approaches in the S&P 500 and DJIA markets. The trading period of the test set is 814 days in which the portfolios generated by the MASAC satisfy all constraints in 576.8 days in S&P 500 and 765.4 days in DJIA, and meet more than 6 out of 7 constraints per day, whereas the best benchmark

TABLE I

THE PERFORMANCE OF WELL-KNOWN APPROACHES AGAINST THE MASAC FRAMEWORK ON THE S&P 500 AND DJIA INDEXES

Markets	**S&P 500**						**DJIA**					
Models	**FD↑**	**CS↑**	**AR(%) ↑**	**MDD(%) ↓**	**SR↑**	**Vol↓**	**FD↑**	**CS↑**	**AR(%) ↑**	**MDD(%) ↓**	**SR↑**	**Vol↓**
EG	0	0.65	20.28	36.75	0.79	0.2352	0	0.87	14.64	19.70	0.81	0.1594
PAMR	0	2.30	-5.40	59.25	-0.17	0.4070	0	2.27	-34.30	79.98	-1.29	0.2783
RMR	0	2.36	-12.01	67.41	-0.29	0.4696	0	2.28	-13.17	63.55	-0.54	0.2733
XPRESS	88	5.29	20.21	38.80	0.60	0.3109	110	6.07	5.55	23.47	0.22	0.1767
GUROBI	42	4.85	20.54	38.88	0.61	0.3120	69	5.80	5.87	22.81	0.24	0.1765
CPLEX	60	4.98	20.23	38.90	0.60	0.3111	84	5.89	5.87	23.43	0.24	0.1766
DPM	0	0.77	16.19	41.27	0.61	0.2415	0	0.96	14.92	18.13	0.82	0.1622
PPN	0	1.22	23.29	38.70	0.85	0.2470	0	0.90	13.90	18.54	0.82	0.1499
RAT	0	0.67	19.53	35.92	0.78	0.2283	0	0.89	14.11	18.84	0.81	0.1528
DT	0	2.97	11.88	45.55	0.37	0.2728	0.2	3.21	2.68	31.32	0.05	0.1829
MASA	0	1.81	18.90	**26.72**	0.82	**0.2102**	0.2	2.14	**16.27**	18.86	**0.90**	0.1617
MASAC	**576.8**	**6.11**	**25.32**	35.33	**0.98**	0.2411	**765.4**	**6.89**	13.98	**14.94**	0.84	**0.1475**

algorithm only satisfies all constraints at most 110 trading days and around 6 constraints per day. This significantly demonstrates the ability of the proposed framework to solve different types of trading constraints at the same time. On the other hand, as the example in S&P 500 market, the MASAC achieves the highest AR at 25.32%, the highest SR at 0.98, and the second-best MDD at 35.33%. The solver-based methods have poor profits due to the lack of ability to estimate future returns while the deep RL-based methods cannot handle multiple constraints. For instance, the MASA focuses on risk management but fails to handle other constraints, achieving the best MDD but lower returns. Moreover, as shown in DJIA, the MASAC satisfies more trading constraints, yet it obtains lower returns since some profitable opportunities are sacrificed. Fig. 3 shows the heatmap of the satisfaction rates of constraints where the higher satisfaction rate is represented by the darker square. The MASAC demonstrates a strong capacity to handle different types of constraints, while other methods usually fail to deal with specific types of constraints.

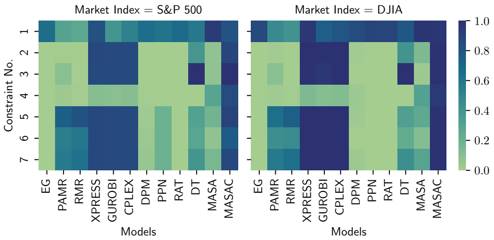

Fig. 3. A Comparison of the Satisfaction Rates of Constraints

V. CONCLUDING REMARKS

Portfolio optimization has been studied for a few decades. Yet the existing approaches mainly focus on maximizing portfolio returns but rarely consider the practical trading constraints, which cannot be adapted to investors with different preferences in real-world trading. To fill the research gap, a **M**ulti-**A**gent and **S**elf-**A**daptive framework for **C**onstrained portfolio optimization, namely the MASAC, is proposed where an RL-based agent and a heuristic-based agent cooperatively generate trading strategies for maximizing overall returns while satisfying all concerned trading requirements. The simulation results reveal that the MASAC well balances investment performance and practical trading requirements in real-world datasets. More importantly, the proposed trading system sheds light on many financial applications and products.

REFERENCES

[1] H. Markowitz and H. Markowitz, *Mean-variance Analysis in Portfolio Choice and Capital Markets*. Pennsylvania: Blackwell, 1990.

[2] L. P. Ling and Y. Dasril, "Portfolio selection strategies in bursa malaysia based on quadratic programming," *Journal of Information System Exploration and Research*, vol. 1, no. 2, 2023.

[3] Y. Fang *et al.*, "Learning multi-agent intention-aware communication for optimal multi-order execution in finance," in *SIGKDD*, 2023.

[4] L. Han, N. Ding *et al.*, "Efficient continuous space policy optimization for high-frequency trading," in *SIGKDD*, 2023, pp. 4112–4122.

[5] W. Han *et al.*, "Select and trade: Towards unified pair trading with hierarchical reinforcement learning," in *SIGKDD*, 2023, pp. 4123–4134.

[6] S. Gao, Y. Wang, and X. Yang, "Stockformer: Learning hybrid trading machines with predictive coding," in *IJCAI*, 2023, pp. 4766–4774.

[7] Z. Wang, B. Huang *et al.*, "Deeptrader: A deep reinforcement learning approach for risk-return balanced portfolio management with market conditions embedding," in *AAAI*, 2021, pp. 643–650.

[8] B. Li and S. C. Hoi, "Online portfolio selection: A survey," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, pp. 1–36, 2014.

[9] D. P. Helmbold *et al.*, "On-line portfolio selection using multiplicative updates," *Mathematical Finance*, vol. 8, no. 4, pp. 325–347, 1998.

[10] B. Li *et al.*, "Pamr: Passive aggressive mean reversion strategy for portfolio selection," *Machine learning*, vol. 87, pp. 221–258, 2012.

[11] D. Huang *et al.*, "Robust median reversion strategy for online portfolio selection," *IEEE TKDE*, vol. 28, no. 9, pp. 2480–2493, 2016.

[12] IBM, "Cplex: A mathematical program solver," 2024. [Online]. Available: https://www.ibm.com/products/ilog-cplex-optimization-studio

[13] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2024. [Online]. Available: https://www.gurobi.com

[14] FICO, "Xpress optimizer reference manual," 2024. [Online]. Available: https://www.fico.com/fico-xpress-optimization/docs/latest/overview.html

[15] Z. Jiang *et al.*, "A deep reinforcement learning framework for the financial portfolio management problem," *arXiv preprint arXiv:1706.10059*, 2017.

[16] Y. Zhang *et al.*, "Cost-sensitive portfolio selection via deep reinforcement learning," *IEEE TKDE*, vol. 34, no. 1, pp. 236–248, 2020.

[17] K. Xu, Y. Zhang *et al.*, "Relation-aware transformer for portfolio policy learning," in *IJCAI*, 2021, pp. 4647–4653.

[18] Z. Li, V. Tam, and K. L. Yeung, "Developing a multi-agent and self-adaptive framework with deep reinforcement learning for dynamic portfolio risk management," in *AAMAS*, 2024.

[19] H. Zhang *et al.*, "Optimizing trading strategies in quantitative markets using multi-agent reinforcement learning," in *ICASSP*, 2024.

[20] S. Almahdi and S. Y. Yang, "A constrained portfolio trading system using particle swarm algorithm and recurrent reinforcement learning," *Expert Systems with Applications*, vol. 130, pp. 145–156, 2019.

[21] V. Tam, "Applying egenet to solve continuous constrained optimization problems: a preliminary report," in *International Conference on Information Intelligence and Systems*. IEEE, 1999, pp. 115–122.