Augmenting Traders with Learning Machines

Ph.D. Thesis Defense

Edoardo Vittori

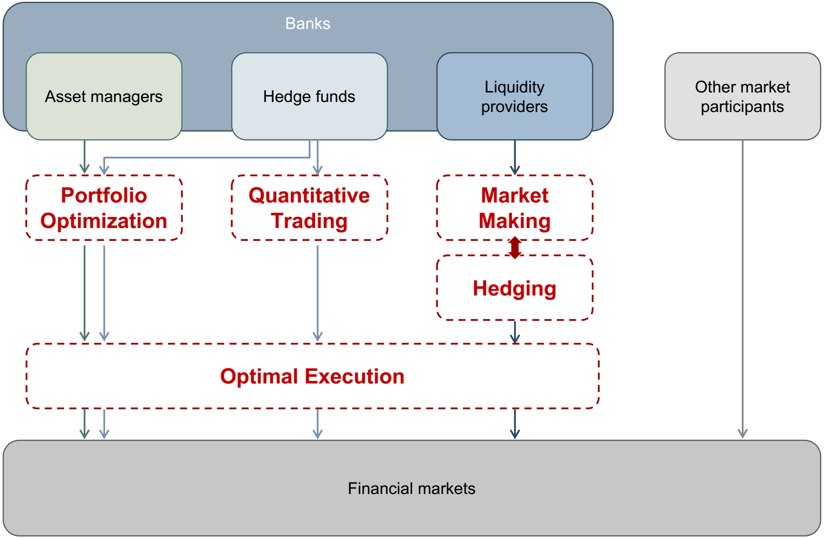
IMI CORPORATE & INVESTMENT BANKING
INTESA SANPAOLO

POLITECNICO MILANO 1863

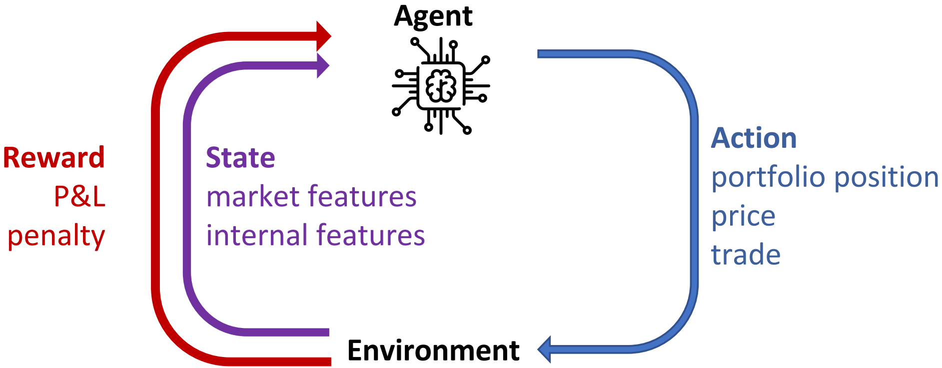
Agenda

1. Introduction

Content Map


Banks
Asset managers
Hedge funds
Liquidity providers
Other market participants
Portfolio Optimization
Quantitative Trading
Market Making
Hedging
Optimal Execution
Financial markets

Trading as a Markov Decision Process (MDP)



Families of Algorithms Considered

	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Online vs offline			
Learning approach			
Observation to action delay			
Policy type			

Families of Algorithms Considered

	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Online vs offline	depends on algorithm		
Learning approach	general policy		
Observation to action delay	small		
Policy type	stationary		

Families of Algorithms Considered

	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Online vs offline	depends on algorithm	online	
Learning approach	general policy	local policy	
Observation to action delay	small	some	
Policy type	stationary	non-stationary	

Families of Algorithms Considered

	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Online vs offline	depends on algorithm	online	online
Learning approach	general policy	local policy	optimizing for next action
Observation to action delay	small	some	small
Policy type	stationary	non-stationary	adversarial

Research Framework

Financial tasks	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Portfolio optimization			
Quantitative trading			
Market making			
Option hedging			
Optimal execution			

Application of Algorithms to Financial Tasks in this Research

Financial tasks	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Portfolio optimization			
Quantitative trading	FQI for FX trading		
Market making	FQI with MFGs for bond dealing		
Option hedging	**Risk averse RL (TRVO) for hedging**		
Optimal execution	FQI and Thompson Sampling		

Application of Algorithms to Financial Tasks in this Research

Financial tasks	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Portfolio optimization			
Quantitative trading	FQI for FX trading	**Open Loop UCT for FX trading**	
Market making	FQI with MFGs for bond dealing		
Option hedging	**Risk averse RL (TRVO) for hedging**		
Optimal execution	FQI and Thompson Sampling		

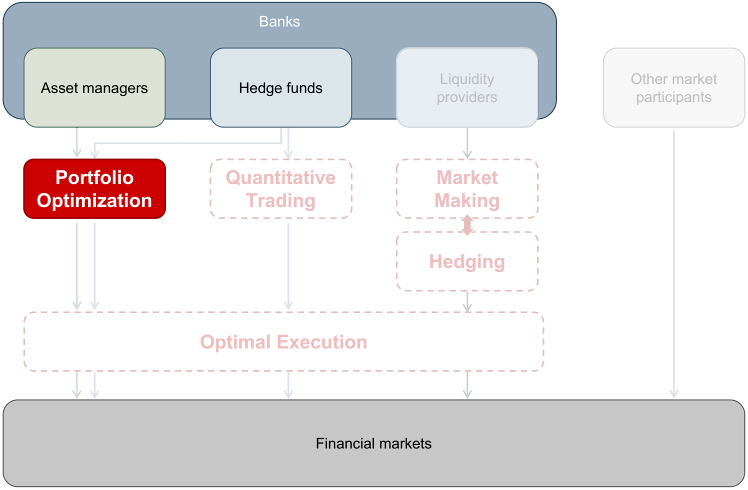
Application of Algorithms to Financial Tasks in this Research

Financial tasks	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Portfolio optimization			Online portfolio optimization
Quantitative trading	FQI for FX trading	Open Loop UCT for FX trading	
Market making	FQI with MFGs for bond dealing		
Option hedging	Risk averse RL (TRVO) for hedging		
Optimal execution	FQI and Thompson Sampling		

Topics of Today's Presentation

Financial tasks	Reinforcement Learning	Monte Carlo Tree Search	Expert Learning
Portfolio optimization			Online portfolio optimization
Quantitative trading	FQI for FX trading	Open Loop UCT for FX trading	
Market making	FQI with MFGs for bond dealing		
Option hedging	Risk averse RL (TRVO) for hedging		
Optimal execution	FQI and Thompson Sampling		

2. Online Portfolio Optimization with Transaction Costs



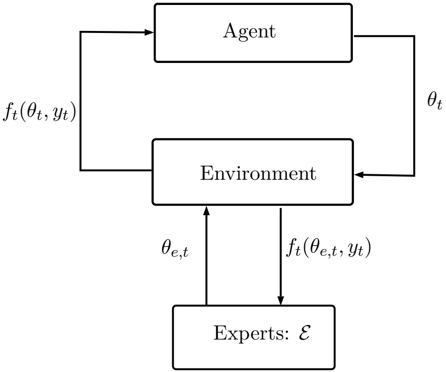
Defining Expert Learning

Expert Learning

1. Agent makes a decision: $\theta_t \in \Theta$, based on suggestions of experts $\mathcal{E}$
2. Environment chooses outcome y_t and loss $f_t(\theta_t, y_t)$
3. Update cumulative loss $L_T = \sum_{t=1}^{T} f_t(\theta_t, y_t)$

Objective

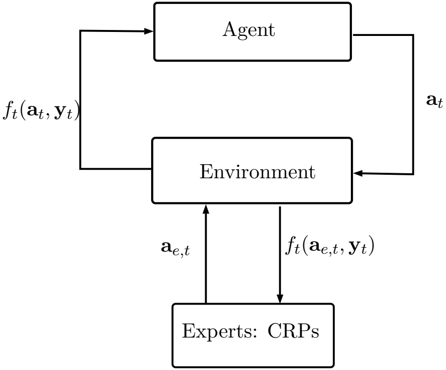
- Regret: $R_T = L_T - \inf_{e \in \mathcal{E}} \sum_{t=1}^{T} f_t(\theta_{e,t}, y_t)$
- No regret: $\frac{R_T}{T} \to 0$



From Expert Learning to Online Portfolio Optimization

Online Portfolio Optimization Setting

- $\mathbf{a}_t \in \Delta_{M-1}$ is the portfolio allocation
- The experts are Constant Rebalancing Portfolios
- $\mathbf{a}^* = \arg\inf_{a \in \Delta_{M-1}} \sum_{t=1}^{T} f_t(\mathbf{a}, \mathbf{y}_t)$ is the Best CRP
- $f_t(\mathbf{a}, \mathbf{y}_t) = -\log(\langle \mathbf{a}, \mathbf{y}_t \rangle)$ is the loss
- $\mathbf{y}_t = \left(\frac{p_{t,1}}{p_{t-1,1}}, \ldots, \frac{p_{t,M}}{p_{t-1,M}} \right)$ are the price relatives



Limitations: no transaction costs

Approaches to Portfolio Optimization

Background

- **Modern Portfolio Optimization**
  [Markowitz, 1952]
  - Calculate variance and correlations
  - Single period
- **Intertemporal CAPM**
  [Merton, 1969]
  - Make assumptions on asset dynamics
  - Multi period
- **Online Portfolio Optimization**
  [Cover and Ordentlich, 1996]
  - Adversarial market
  - Multi period

Approaches to Portfolio Optimization

Background

- Modern Portfolio Optimization
  [Markowitz, 1952]
  - Calculate variance and correlations
  - Single period
- Intertemporal CAPM
  [Merton, 1969]
  - Make assumptions on asset dynamics
  - Multi period
- Online Portfolio Optimization
  [Cover and Ordentlich, 1996]
  - Adversarial market
  - Multi period

Main contributions

Dealing with Transaction Costs in Portfolio Optimization: Online Gradient Descent with Momentum
[Vittori et al., 2020a]

- Keeping transaction costs under control in OPO
- Definition of a algorithm: OGDM with total regret guarantees

Total Regret: Adding Transaction Costs

Total Regret

$$R_T^C = \underbrace{\sum_{t=1}^{T} f_t(\mathbf{a}_t, \mathbf{y}_t) - \inf_{a \in \Delta_{M-1}} \sum_{t=1}^{T} f_t(\mathbf{a}, \mathbf{y}_t)}_{R_T\text{: standard regret}} + \underbrace{\gamma \sum_{t=1}^{T} ||\mathbf{a}_t - \mathbf{a}_{t-1}||_1}_{C_T\text{: transaction costs}}$$

γ is the proportional transaction rate for buying and selling stocks

Online Gradient Descent with Momentum

Algorithm 1 OGDM in OPO with Transaction Costs

Require: learning rate sequence $\{\eta_1, \ldots, \eta_T\}$, momentum parameter sequence $\{\lambda_1, \ldots, \lambda_T\}$

Set $\mathbf{a}_1 \leftarrow \frac{1}{M}\mathbf{1}$
for $t \in \{1, \ldots, T\}$ **do**
  Select $\mathbf{a}_{t+1} \leftarrow \Pi_{\Delta_{M-1}} \left(\mathbf{a}_t + \eta_t \frac{\mathbf{y}_t}{\langle \mathbf{y}_t, \mathbf{a}_t \rangle} - \frac{\lambda_t}{2}(\mathbf{a}_t - \mathbf{a}_{t-1}) \right)$
  Observe $\mathbf{y}_{t+1}$ from the market
  Get wealth $\log(\langle \mathbf{y}_{t+1}, \mathbf{a}_{t+1} \rangle) - \gamma ||\mathbf{a}_{t+1} - \mathbf{a}_t||_1$
end for

Total Regret

$$R_T^C \leq \mathcal{O}(\sqrt{T})$$

Vittori, E., Bernasconi De Luca, M., Trovò, F., and Restelli, M. (2020). **Dealing with Transaction Costs in Portfolio Optimization: Online Gradient Descent with Momentum**. *ICAIF*.

Comparison with State of the Art in OPO

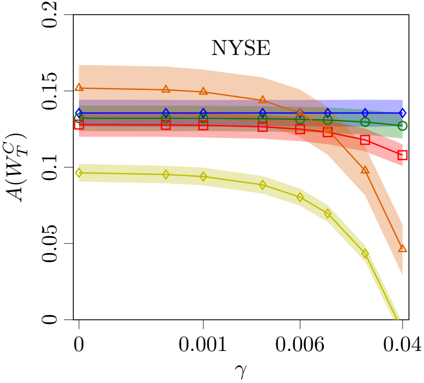
Online Portfolio Optimization

- Universal Portfolios (U_CP) [Kalai and Vempala, 2002]
- Online Newton Step (ONS) [Agarwal et al., 2006]
- Online Lazy Updates (OLU) [Das et al., 2013]

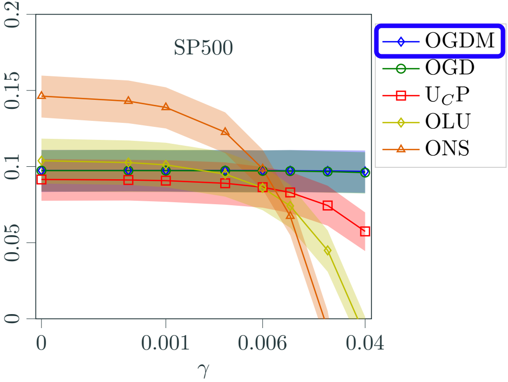
	Algorithm type			
Metric	**OGDM**	**U_CP**	**OLU**	**ONS**
R_T	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\log T)$	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\log T)$
R_T^C	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\log T)$	$\mathcal{O}(T)$	-
Complexity	$\Theta(M)$	$\Theta(T^M)$	$\Theta(M)$	$\Theta(M^2)$

Experimental Results: Average APY

Average results NYSE dataset



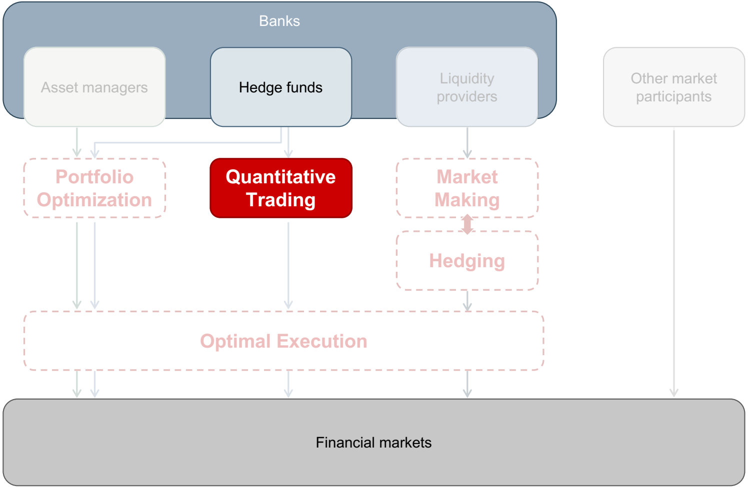
Average results SP500 dataset



Average Annual Percentage Yield $A(W_T)$ computed on the wealth $W_T^C(\mathbf{a}_{1:T}, \mathbf{y}_{1:T})$: $A(W_T) = W_T^{250/T} - 1$

Vittori, E., Bernasconi De Luca, M., Trovò, F., and Restelli, M. (2020). **Dealing with Transaction Costs in Portfolio Optimization: Online Gradient Descent with Momentum**. *ICAIF*.

3. Quantitative Trading with MCTS



Quantitative Trading

Trading: *a sequential decision process in which at each round $t \in \{1, \ldots, T\}$ over a trading horizon $T \in \mathbb{N}$, a trader decides whether to go long, short or stay flat with respect to an asset to maximize her wealth*

MDP Configuration

- $a_t \in \{-1, 0, 1\}$
- $s_t = ([P_{t-w}, ..., P_t], a_{t-1}, t)$
- $r_{t+1} = \underbrace{a_t(P_{t+1} - P_t)}_{\text{market movement}} - \underbrace{\frac{\text{bid-ask}}{2}|a_t - a_{t-1}|}_{\text{transaction costs}}$

Approaches to Trading

Background

- Practitioner approach
  - Technical analysis
  - Macro-economic analysis
- Supervised learning approach
  [Baba and Kozaki, 1992]
  - Forecast asset prices
  - Derive trade
  - Hard to incorporate market frictions
- Reinforcement Learning approach
  [Moody and Saffell, 2001]
  - Integrate prediction and action
  - Simple to include market frictions

Approaches to Trading

Background

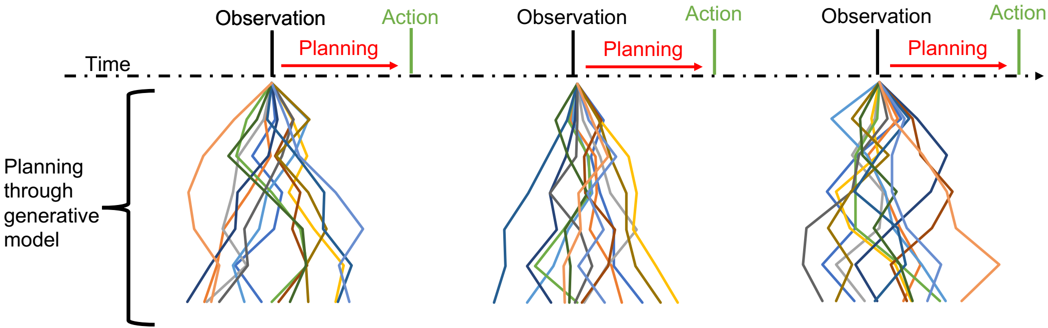
- Practitioner approach
  - Technical analysis
  - Macro-economic analysis
- Supervised learning approach
  [Baba and Kozaki, 1992]
  - Forecast asset prices
  - Derive trade
  - Hard to incorporate market frictions
- Reinforcement Learning approach
  [Moody and Saffell, 2001]
  - Integrate prediction and action
  - Simple to include market frictions

Main contributions

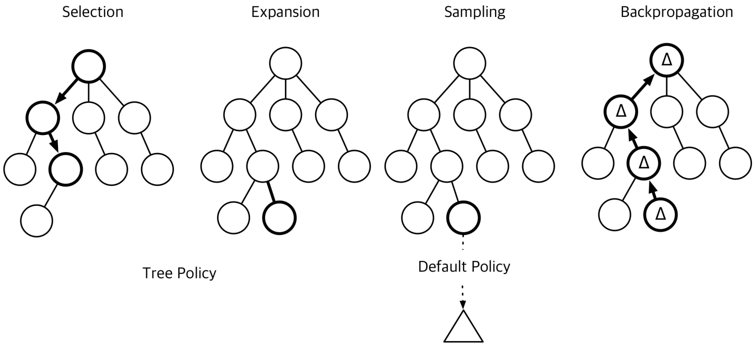
Monte Carlo Tree Search for Trading and Hedging
[Vittori et al., 2021]

- Use of Open Loop MCTS for single currency FX trading

Monte Carlo Tree Search (MCTS)

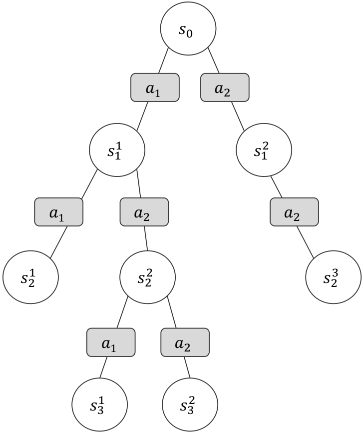


Upper Confidence Tree (UCT) [Kocsis and Szepesvári, 2006]

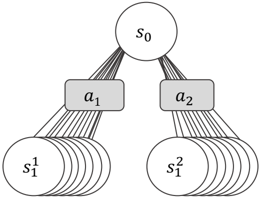


- Selection using UCB1 $a_n = \arg\max_{i=1..K} \overline{X}_{i,T_i(n-1)} + C\sqrt{\frac{2\log n}{T_i(n-1)}}$
- Convergence to the optimal solution in deterministic environments

UCT in deterministic environments



UCT in continuous stochastic environments



Open Loop UCT [Lecarpentier et al., 2018]

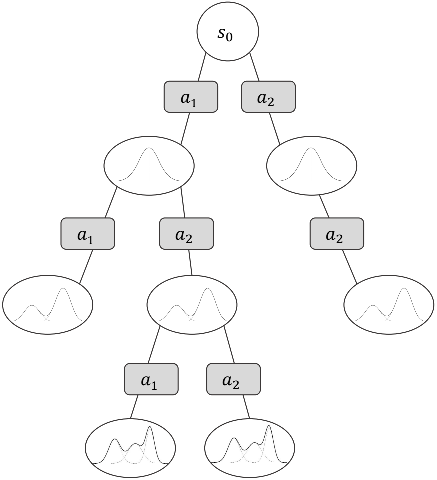
- Nodes are distributions over states
- Open-loop value of action sequence τ:

$$V_{OL}(s,\tau) = \mathbb{E}\left[\sum_{t=1}^{m} \gamma^t r_t \middle| s_0 = s, a_t \in \tau\right]$$

- Open-loop value of a node $\mathcal{N}_{d,i}$:

$$\mathcal{V}\left(\mathcal{N}_{d,i}\right) = \mathop{\mathbb{E}}_{s\sim\mathcal{P}(\cdot|s_0,\tau_{d,i})}\left[V^*_{OL}(s)\right]$$

where $V^*_{OL}(s) = \max_{\tau\in\mathcal{A}^m} V_{OL}(s,\tau)$



Q-learning Backpropagation

- **Standard** Backpropagation

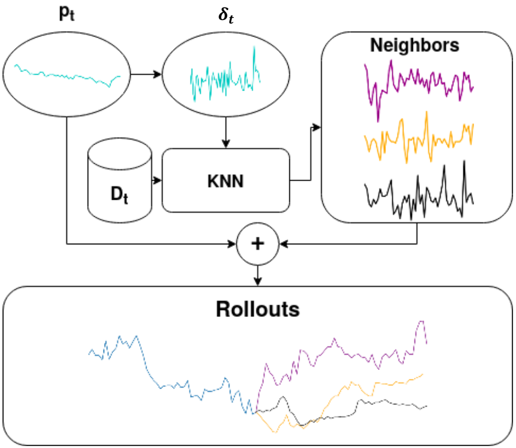
$$\mathcal{Q}_t\left(\mathcal{N}_{d,i}, a\right) = (1 - \frac{1}{N})\mathcal{Q}_t\left(\mathcal{N}_{d,i}, a\right) + \frac{1}{N}(r_t + \gamma \mathcal{V}_t\left(\mathcal{N}_{d+1,j}\right))$$

- **Temporal Difference** Backpropagation, based on the Q-Learning update rule [Vodopivec et al., 2017]

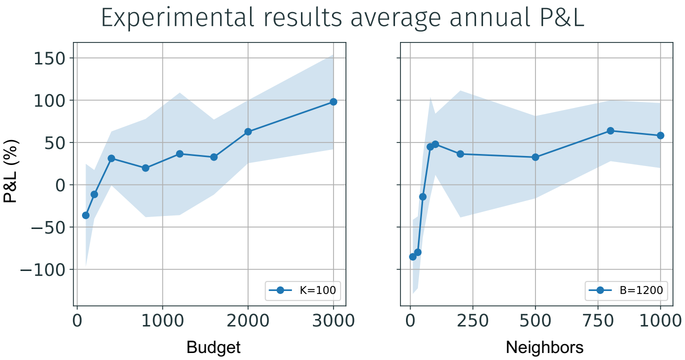
$$\mathcal{Q}_t\left(\mathcal{N}_{d,i}, a\right) = (1 - \beta)\mathcal{Q}_t\left(\mathcal{N}_{d,i}, a\right) + \beta\left(r_t + \gamma \max_{a'} \mathcal{Q}_t\left(\mathcal{N}_{d+1,j}, a'\right)\right)$$

Clustering generative model

1. Start from the current price window $w_t = (P_{t-M}, \ldots, P_{t-1})$
2. Extract window of returns $\delta_t = \frac{P_t - P_{t-1}}{P_{t-1}}$, $\boldsymbol{\delta}_t = (\delta_{t-M}, \ldots, \delta_{t-1})$
3. Find the K nearest neighbors of $\boldsymbol{\delta}_t$ in the historical dataset D
4. Use the neighbors to project future asset prices



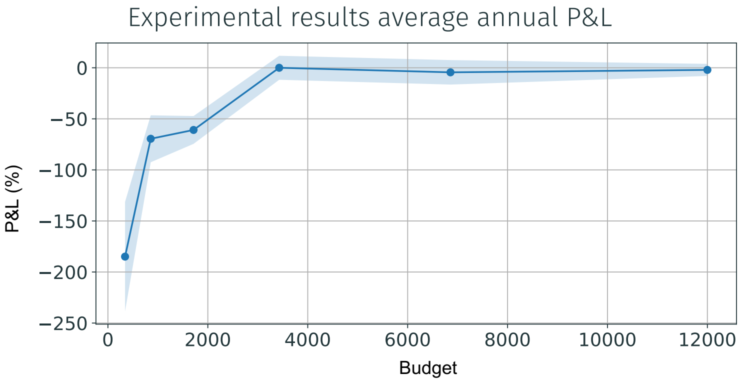
Experimental Results Trading EURUSD FX without Transaction Costs



Annualized average P&L with no transaction costs, as a function of the search budget and the numbers of neighbors. Average over 50 runs, 95% confidence intervals

Vittori, E., Likmeta A., and Restelli, M. (2021). **Monte carlo tree search for trading and hedging**. *ICAIF*.

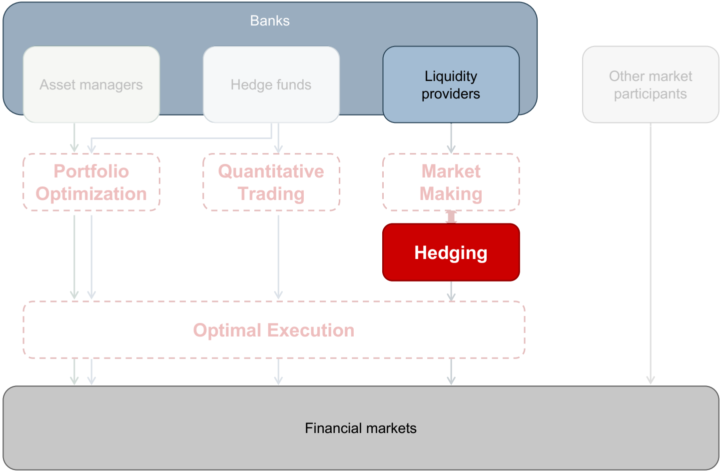
Experimental Results Trading EURUSD FX with Transaction Costs



Annualized average P&L with transaction costs (10^{-5}) as a function of the search budget, $K = 100$. Average over 50 runs, 95% confidence intervals

Vittori, E., Likmeta A., and Restelli, M. (2021). **Monte carlo tree search for trading and hedging**. *ICAIF*.

4. Option Hedging with Risk-Averse RL

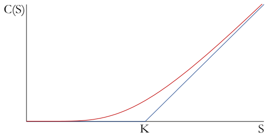


Vanilla options: contracts that offer the buyer the right to buy or sell a certain amount of the underlying asset at a predefined price at a certain future time

Option hedging: *a sequential decision process in which at each round $t \in \{1, \dots, T\}$ over the life of the option $T \in \mathbb{N}$, a trader decides how much to hold of the underlying instrument to minimize the price swings caused by the option*

Option Hedging as an MDP

- $a_t \in [0, 1]$: current hedge portfolio
- $s_t = [S_t, C_t, \frac{\partial C(S_t)}{\partial S}, a_{t-1}]$
- $r_{t+1} = \underbrace{C_{t+1} - C_t}_{\text{option variation}} \underbrace{- a_t \cdot (S_{t+1} - S_t)}_{\text{market movement}} - \underbrace{c(a_t - a_{t-1})}_{\text{transact. costs}}$



Background

- **Classical approach**
  [Black and Scholes, 1973]
  - Model the market as GBM
  - Assume continuous time hedging
  - Assume no market frictions
  - Solve resulting PDE
- **Reinforcement Learning approach**
  [Kolm and Ritter, 2019]
  - Collect/simulate data
  - Learn to hedge

Approaches to Option Hedging

Background

- Classical approach
  [Black and Scholes, 1973]
  - Model the market as GBM
  - Assume continuous time hedging
  - Assume no market frictions
  - Solve resulting PDE
- Reinforcement Learning approach
  [Kolm and Ritter, 2019]
  - Collect/simulate data
  - Learn to hedge

Main contributions

Option Hedging with Risk Averse RL
[Vittori et al., 2020b]

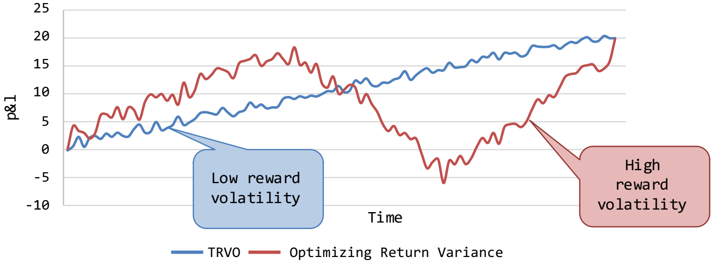
- Use of the risk-averse policy search RL algorithm: TRVO

Trust Region Volatility Optimization (TRVO)

- Reward volatility

$$\nu_\pi^2 = (1-\gamma) \underset{\substack{s_0 \sim \mu \\ a_t \sim \pi(\cdot|s_t) \\ s_{t+1} \sim \mathcal{P}(\cdot|s_t, a_t)}}{\mathbb{E}} \left[\sum_{t=0}^{\infty} \gamma^t \left(\mathcal{R}(s_t, a_t) - J_\pi \right)^2 \right]$$

- Mean-volatility objective $\eta_\pi = J_\pi - \lambda \nu_\pi^2$



Bisi, L., Sabbioni, L., Vittori, E., Papini, M., and Restelli, M. (2020). **Risk-averse trust region optimization for reward-volatility reduction**. *IJCAI*.

Financial Environment

Vanilla call option

- time to maturity $=$ 60 days
- unitary notional
- implied volatility $= 20\%$
- interest rates $= 0$
- $K = S_0 = 100$
- starting price (ATM) option ~ 3.24
- starting delta $= 0.5$

Simulated underlying

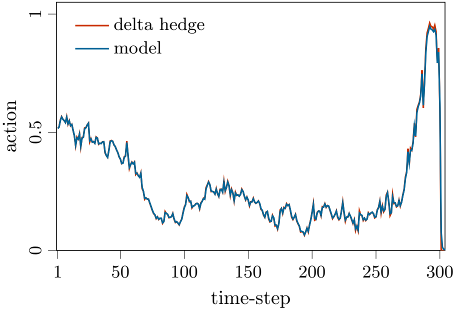
- GBM
- no drift
- volatility $= 20\%$
- $S_0 = 100$
- 5 time steps per day
- bid ask spread $= 0.1$

training on 10k episodes and testing on 2k episodes

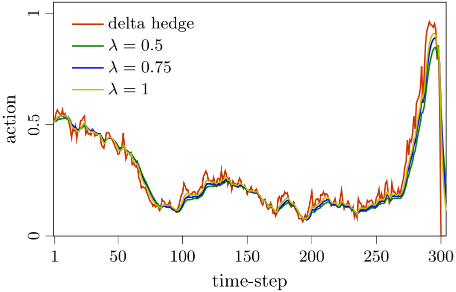
Vittori, E., Trapletti, M., and Restelli, M. (2020). **Option Hedging with Risk Averse Reinforcement Learning**. *ICAIF*.

Experimental Results, Action per Time-step

Results without transcation costs



Results with transcation costs

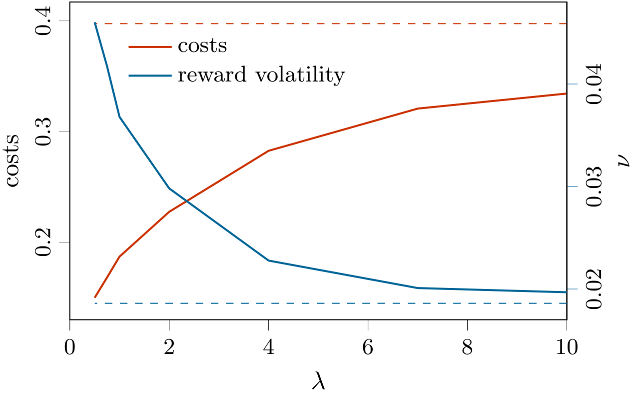


- delta hedge with no transaction costs $\rightarrow$ average P&L $\sim$ 0, volatility $\sim$ 0.16
- delta hedge with transaction costs $\rightarrow$ average P&L $\sim$ -0.3, volatility $\sim$ 0.18

Vittori, E., Trapletti, M., and Restelli, M. (2020). **Option Hedging with Risk Averse Reinforcement Learning**. *ICAIF*.

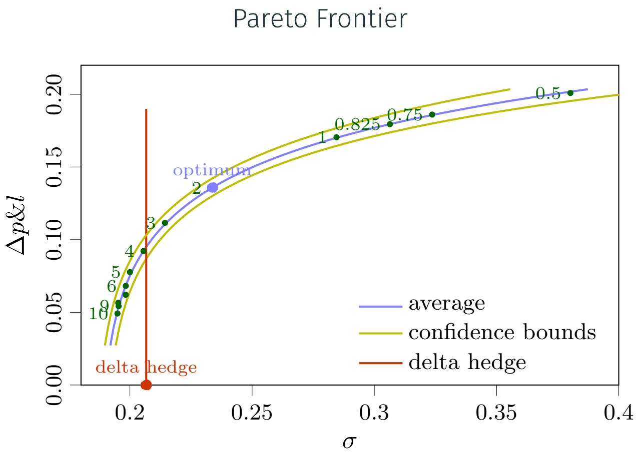
Experimental Results with Transaction Costs

Costs vs Risk changing λ



Vittori, E., Trapletti, M., and Restelli, M. (2020). **Option Hedging with Risk Averse Reinforcement Learning**. *ICAIF*.

Experimental Results with Transaction Costs



Vittori, E., Trapletti, M., and Restelli, M. (2020). **Option Hedging with Risk Averse Reinforcement Learning**. *ICAIF*.

5. Conclusions

Conclusions

Today's Topics

- **Online portfolio optimization**
  - Controlling transaction costs in OPO
- **Quantitative trading**
  - FX trading using Open Loop UCT
- **Option hedging**
  - Equity option hedging using TRVO

Conclusions

Today's Topics

- **Online portfolio optimization**
  - Controlling transaction costs in OPO
- **Quantitative trading**
  - FX trading using Open Loop UCT
- **Option hedging**
  - Equity option hedging using TRVO

Final Remarks

- Major financial tasks in the Capital Markets modelled as MDPs
- Broad applicability of RL based techniques to financial problems
- Data driven approaches without explicit modelling assumptions

Q&A

CONTACTS

edoardo.vittori@polimi.it

edoardo-vittori

References i

[Agarwal et al., 2006] Agarwal, A., Hazan, E., Kale, S., and Schapire, R. (2006).
Algorithms for portfolio management based on the newton method.
In *ICML*.

[Almgren and Chriss, 2001] Almgren, R. and Chriss, N. (2001).
Optimal execution of portfolio transactions.
Journal of Risk, 3:5–40.

[Avellaneda and Stoikov, 2008] Avellaneda, M. and Stoikov, S. (2008).
High-frequency trading in a limit order book.
Quantitative Finance, 8(3):217–224.

[Baba and Kozaki, 1992] Baba, N. and Kozaki, M. (1992).
An intelligent forecasting system of stock price using neural networks.
In *IJCNN*, volume 1.

[Bernasconi de Luca et al., 2021] Bernasconi de Luca, M., Vittori, E., Trovò, F., and Restelli, M. (2021).
Conservative online convex optimization.
In *ECML*.

[Bisi et al., 2020] Bisi, L., Sabbioni, L., Vittori, E., Papini, M., and Restelli, M. (2020).
Risk-averse trust region optimization for reward-volatility reduction.
In *IJCAI. Special Track on AI in FinTech.*

References ii

[Black and Scholes, 1973] Black, F. and Scholes, M. (1973).
The pricing of options and corporate liabilities.
Journal of political economy, 81(3):637–654.

[Boyd et al., 2017] Boyd, S., Busseti, E., Diamond, S., Kahn, R. N., Koh, K., Nystrup, P., and Speth, J. (2017).
Multi-period trading via convex optimization.
arXiv preprint.

[Briola et al., 2021] Briola, A., Turiel, J., Marcaccioli, R., and Aste, T. (2021).
Deep reinforcement learning for active high frequency trading.
arXiv preprint.

[Buehler et al., 2019] Buehler, H., Gonon, L., Teichmann, J., and Wood, B. (2019).
Deep hedging.
Quantitative Finance, pages 1–21.

[Byrd et al., 2019] Byrd, D., Hybinette, M., and Balch, T. H. (2019).
Abides: Towards high-fidelity market simulation for ai research.
arXiv preprint.

[Cannelli et al., 2020] Cannelli, L., Nuti, G., Sala, M., and Szehr, O. (2020).
Hedging using reinforcement learning: Contextual k-armed bandit versus q-learning.
arXiv preprint.

References iii

[Cao et al., 2019] Cao, J., Chen, J., Hull, J. C., and Poulos, Z. (2019).
Deep hedging of derivatives using reinforcement learning.
Available at SSRN.

[Cover and Ordentlich, 1996] Cover, T. and Ordentlich, E. (1996).
Universal portfolios with side information.
IEEE Transactions on Information Theory, 42(2):348–363.

[Das et al., 2013] Das, P., Johnson, N., and Banerjee, A. (2013).
Online lazy updates for portfolio selection with transaction costs.
In *AAAI*.

[Duchi et al., 2011] Duchi, J., Hazan, E., and Singer, Y. (2011).
Adaptive subgradient methods for online learning and stochastic optimization.
Journal of machine learning research, 12(7).

[Ernst et al., 2005] Ernst, D., Geurts, P., and Wehenkel, L. (2005).
Tree-based batch mode reinforcement learning.
JMLR, 6(Apr):503–556.

[Ganesh et al., 2019] Ganesh, S., Vadori, N., Xu, M., Zheng, H., Reddy, P., and Veloso, M. (2019).
Reinforcement learning for market making in a multi-agent dealer market.
arXiv preprint.

[Gârleanu and Pedersen, 2013] Gârleanu, N. and Pedersen, L. H. (2013).
Dynamic trading with predictable returns and transaction costs.
The Journal of Finance, 68(6):2309–2340.

[Guéant and Manziuk, 2019] Guéant, O. and Manziuk, I. (2019).
Deep reinforcement learning for market making in corporate bonds: beating the curse of dimensionality.
Applied Mathematical Finance, 26(5):387–452.

[Guo et al., 2019] Guo, X., Hu, A., Xu, R., and Zhang, J. (2019).
Learning mean-field games.
arXiv preprint.

[Halperin, 2019] Halperin, I. (2019).
The qlbs q-learner goes nuqlear: fitted q iteration, inverse rl, and option portfolios.
Quantitative Finance, pages 1–11.

[Hazan, 2019] Hazan, E. (2019).
Introduction to online convex optimization.
arXiv preprint.

[Hazan et al., 2007] Hazan, E., Agarwal, A., and Kale, S. (2007).
Logarithmic regret algorithms for online convex optimization.
MACH LEARN, 69:169–192.

References v

[Hendricks and Wilcox, 2014] Hendricks, D. and Wilcox, D. (2014).
A reinforcement learning extension to the almgren-chriss framework for optimal trade execution.
In *CIFEr*, pages 457–464. IEEE.

[Hoi et al., 2021] Hoi, S. C., Sahoo, D., Lu, J., and Zhao, P. (2021).
Online learning: A comprehensive survey.
Neurocomputing, 459:249–289.

[Kalai and Vempala, 2002] Kalai, A. and Vempala, S. (2002).
Efficient algorithms for universal portfolios.
J MACH LEARN RES, 3(Nov):423–440.

[Karpe et al., 2020] Karpe, M., Fang, J., Ma, Z., and Wang, C. (2020).
Multi-agent reinforcement learning in a realistic limit order book market simulation.
arXiv preprint.

[Kocsis and Szepesvári, 2006] Kocsis, L. and Szepesvári, C. (2006).
Bandit based monte-carlo planning.
In *ECML*.

[Kolm and Ritter, 2019] Kolm, P. N. and Ritter, G. (2019).
Dynamic replication and hedging: A reinforcement learning approach.
The Journal of Financial Data Science, 1(1):159–171.

[Kolm and Ritter, 2020] Kolm, P. N. and Ritter, G. (2020).
Modern perspectives on reinforcement learning in finance.
Available at SSRN.

[Lecarpentier et al., 2018] Lecarpentier, E., Infantes, G., Lesire, C., and Rachelson, E. (2018).
Open loop execution of tree-search algorithms.
In *IJCAI*.

[Li and Hoi, 2014] Li, B. and Hoi, S. (2014).
Online portfolio selection: A survey.
ACM COMPUT SURV, 46(3):35.

[Li et al., 2018] Li, B., Wang, J., Huang, D., and Hoi, S. (2018).
Transaction cost optimization for online portfolio selection.
QUANT FINANC, 18(8):1411–1424.

[Li et al., 2012] Li, B., Zhao, P., Hoi, S., and Gopalkrishnan, V. (2012).
Pamr: Passive aggressive mean reversion strategy for portfolio selection.
MACH LEARN, 87(2):221–258.

[Lin and Beling, 2020] Lin, S. and Beling, P. A. (2020).
An end-to-end optimal trade execution framework based on proximal policy optimization.
In *IJCAI*, pages 4548–4554.

[Liu et al., 2020] Liu, X.-Y., Yang, H., Chen, Q., Zhang, R., Yang, L., Xiao, B., and Wang, C. D. (2020).
Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance.
arXiv preprint.

[Markowitz, 1952] Markowitz, H. (1952).
Portfolio selection.
The journal of finance, 7(1):77–91.

[Meng and Khushi, 2019] Meng, T. L. and Khushi, M. (2019).
Reinforcement learning in financial markets.
Data, 4(3):110.

[Merton, 1969] Merton, R. C. (1969).
Lifetime portfolio selection under uncertainty: The continuous-time case.
The review of Economics and Statistics, pages 247–257.

[Mnih et al., 2013] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013).
Playing atari with deep reinforcement learning.
arXiv preprint arXiv:1312.5602.

[Moody and Saffell, 2001] Moody, J. and Saffell, M. (2001).
Learning to trade via direct reinforcement.
IEEE transactions on neural Networks, 12(4):875–889.

References viii

[Nevmyvaka et al., 2006] Nevmyvaka, Y., Feng, Y., and Kearns, M. (2006).
Reinforcement learning for optimized trade execution.
In *ICML*, pages 673–680.

[Ning et al., 2018] Ning, B., Lin, F. H. T., and Jaimungal, S. (2018).
Double deep q-learning for optimal execution.
arXiv preprint.

[Ritter, 2017] Ritter, G. (2017).
Machine learning for trading.
Available at SSRN.

[Riva et al., 2021] Riva, A., Bisi, L., Sabbioni, L., Liotet, P., Vittori, E., Trapletti, M., Pinciroli, M., and Restelli, M. (2021).
Learning fx trading strategies with fqi and persistant actions.
In *ICAIF*.

[Schulman et al., 2015] Schulman, J., Levine, S., Abbeel, P., Jordan, M. I., and Moritz, P. (2015).
Trust region policy optimization.
In *ICML*, volume 37, pages 1889–1897.

[Schulman et al., 2017] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017).
Proximal policy optimization algorithms.
CoRR, abs/1707.06347.

[Silver et al., 2017] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. (2017).
Mastering the game of go without human knowledge.
nature, 550(7676):354–359.

[Spooner et al., 2018] Spooner, T., Fearnley, J., Savani, R., and Koukorinis, A. (2018).
Market making via reinforcement learning.
arXiv preprint.

[Spooner and Savani, 2020] Spooner, T. and Savani, R. (2020).
Robust market making via adversarial reinforcement learning.
IJCAI.

[Streeter and McMahan, 2012] Streeter, M. and McMahan, H. B. (2012).
No-regret algorithms for unconstrained online convex optimization.
arXiv preprint arXiv:1211.2260.

[Théate and Ernst, 2021] Théate, T. and Ernst, D. (2021).
An application of deep reinforcement learning to algorithmic trading.
Expert Systems with Applications, 173:114632.

References x

[Vittori et al., 2020a] Vittori, E., Bernasconi de Luca, M., Trovò, F., and Restelli, M. (2020a).
Dealing with transaction costs in portfolio optimization: Online gradient descent with momentum.
In *ICAIF*.

[Vittori et al., 2021] Vittori, E., Likmeta, A., and Restelli, M. (2021).
Monte carlo tree search for trading and hedging.
In *ICAIF*.

[Vittori et al., 2020b] Vittori, E., Trapletti, M., and Restelli, M. (2020b).
Option hedging with risk averse reinforcement learning.
In *ICAIF*.

[Vodopivec et al., 2017] Vodopivec, T., Samothrakis, S., and Ster, B. (2017).
On monte carlo tree search and reinforcement learning.
Journal of Artificial Intelligence Research, 60:881–936.

[Watkins, 1989] Watkins, C. J. C. H. (1989).
Learning from delayed rewards.
PhD thesis, King's College, Cambridge.

[Williams, 1992] Williams, R. J. (1992).
Simple statistical gradient-following algorithms for connectionist reinforcement learning.
Machine learning, 8(3-4):229–256.

[Zinkevich, 2003] Zinkevich, M. (2003).
Online convex programming and generalized infinitesimal gradient ascent.
In *ICML*.

A. Contributions and Challenges

Main Contributions

Main Contributions

- **Online portfolio optimization**
  - Controlling transaction costs in OPO
- **Quantitative trading**
  - FX trading using Open Loop UCT
  - Two currency FX trading using FQI
- **Bond market making**
  - Mean Field Games and FQI
- **Option hedging**
  - Equity option hedging using TRVO
  - Credit option hedging using TRVO
- **Optimal execution**
  - Using TS to adapt to the nonstationarity of the markets

Final Remarks

- Major financial tasks in the Capital Markets sector can be modelled as MDPs
- Broad applicability of RL based techniques to financial problems
- Data driven approaches without explicit modelling assumptions

Current Challenges in Applying RL

- Acquisition of training data
  - Simulation via stochastic models
  - GANs or other advanced ML approaches
- Non-stationarity of the financial markets
  - Market regimes
  - Rare events
- Low signal to noise ratio
  - Control frequency
  - Data processing
- Resistance to trust a completely autonomous trading agent

Future Works I

- Online Portfolio Optimization
  - Evaluate the feasibility of using in a high frequency trading framework
- Quantitative Trading
  - Expand feature set in state, including both microstructural order book facts and possible predictive signals
  - Expand to n asset scenario
- Hedging
  - Expand to hedging of a portfolio of derivatives
- Market Making
  - Use real data or market simulators in order to introduce realism
  - Combine with hedging
- Optimal Execution
  - Improve and generalize the approach
  - Combine with trading

Future Works II

- Reinforcement Learning
  - Dealing with non-stationarity
  - Optimal control frequency
- Monte Carlo Tree Search
  - Extend algorithms such as Alphazero [Silver et al., 2017] to be compatible with continuous stochastic states
  - Improve the generative model
- Expert Learning
  - Analyze potential applications in high frequency scenarios

B. RL Fundamentals

Reinforcement Learning Intro

- Returns

$$G(\tau) = \sum_{t=0}^{\infty} \gamma^t R_t$$

- Action-Value function

$$Q_\pi(s, a) = \mathop{\mathbb{E}}_{\tau \sim \pi}[G(\tau) | s_0 = s, a_0 = a]$$

- Objective

$$J = \max_\pi \mathop{\mathbb{E}}_{\tau \sim \pi}[G(\tau))]$$

RL: Value Based & Policy Search

- **Value based** learn the action-value function

$$Q_\pi(s,a) = \mathop{\mathbb{E}}_{\tau\sim\pi}[G(\tau)|s_0 = s, a_0 = a]$$
$$= r(s,a) + \gamma \mathop{\mathbb{E}}_{\substack{a'\sim\pi \\ s'\sim P}}[Q(s',a')]$$

Bellman Equation

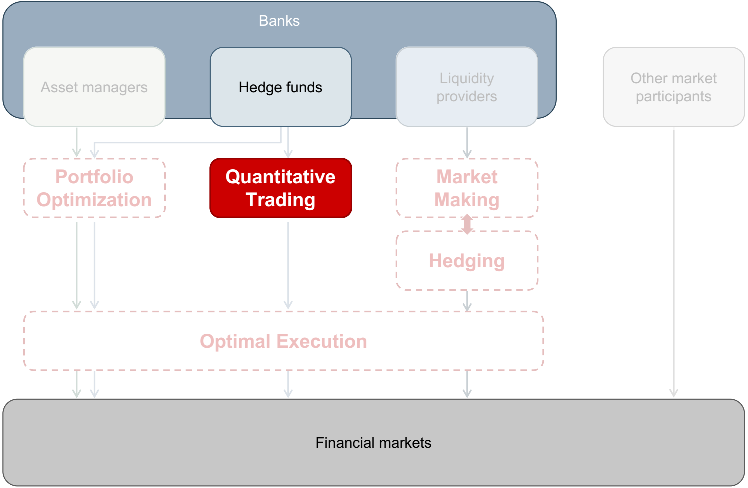
- Examples
  - Q-Learning [Watkins, 1989]
  - FQI [Ernst et al., 2005]
  - DQN [Mnih et al., 2013]

- **Policy search** move in the policy space using experience

$$\nabla_\theta J(\pi_\theta) = \mathop{\mathbb{E}}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) G(\tau)\right]$$

- Examples
  - REINFORCE [Williams, 1992]
  - TRPO [Schulman et al., 2015]
  - PPO [Schulman et al., 2017]

C. Quantitative Trading with FQI



Approaches to Trading

Background

- **Practitioner approach**
  - Technical analysis
  - Macro-economic analysis
- **Supervised learning approach**
  [Baba and Kozaki, 1992]
  - Forecast asset prices
  - Derive trade
  - Hard to incorporate market frictions
- **Reinforcement Learning approach**
  [Moody and Saffell, 2001]
  - Integrate prediction and action
  - Simple to include market frictions

Approaches to Trading

Background

- Practitioner approach
  - Technical analysis
  - Macro-economic analysis
- Supervised learning approach [Baba and Kozaki, 1992]
  - Forecast asset prices
  - Derive trade
  - Hard to incorporate market frictions
- Reinforcement Learning approach [Moody and Saffell, 2001]
  - Integrate prediction and action
  - Simple to include market frictions

Main contributions

Learning FX Trading Strategies with FQI and Persistent Actions [Riva et al., 2021]

- Use of FQI for FX multi-currency trading

Fitted Q-Iteration [Ernst et al., 2005]

$$\mathcal{D} = \{(s_k, a_k, r_k, s'_k)|k = 1, \dots, |\mathcal{D}|\}$$

Algorithm 2 Fitted Q Iteration Algorithm

Require: $\hat{Q}_0(s, a) \leftarrow 0 \ \forall s \in \mathcal{S}, a \in \mathcal{A}$, number of iterations J, and load dataset $\mathcal{D}$

1: **for** $j \in [J]$ **do**

2: $\quad \hat{Q}_{j+1} = \arg\min_{f \in \mathcal{F}} \sum_{s,a,r,s' \in D} \left(f(s, a) - r - \gamma \max_{a \in \mathcal{A}} \hat{Q}_j(s', a) \right)^2$

3: **end for**

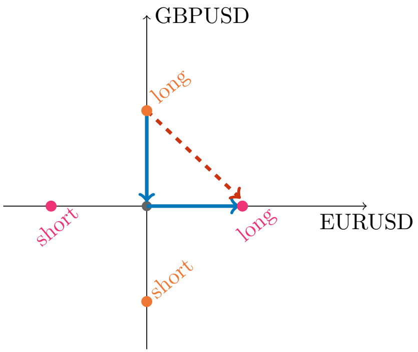
4: Return $\hat{Q}_J$

$\hat{Q}$ as extra-tree regressors $\rightarrow$ min-split tuning

Two Currency Model [Riva et al., 2021]

Two currency model definition

- Two FX pairs with common base currency
- 5 actions: $a_t \in \{1, 2, 3, 4, 5\}$
- Portfolio exposure to one FX pair at a time
- Fixed traded amount of base currency: $\$100k$
- Fixed transaction costs: bid-ask $= \$2 \cdot 10^{-5}$
- Doubled costs for certain trades



Model Assumptions

Trading assumptions

- Episode = Trading Day = 08:00-18:00 CET
- Close any position end of day

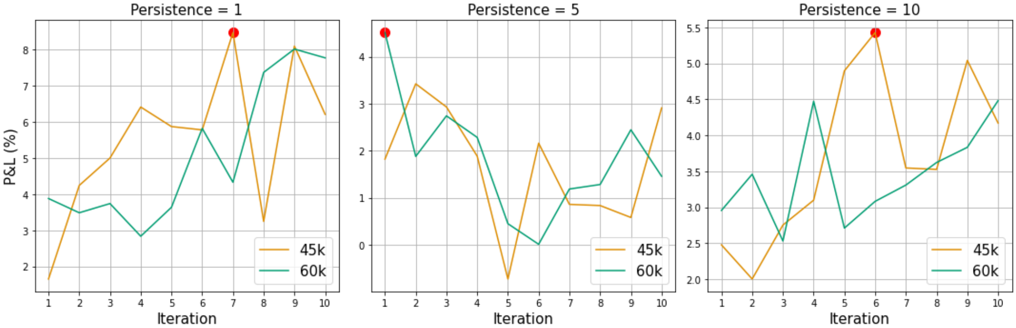
Training and testing settings

- Training set: 2017 - 2018
- Validation set: 2019
- Test set: 2020
- Training algorithm: FQI

MDP assumptions

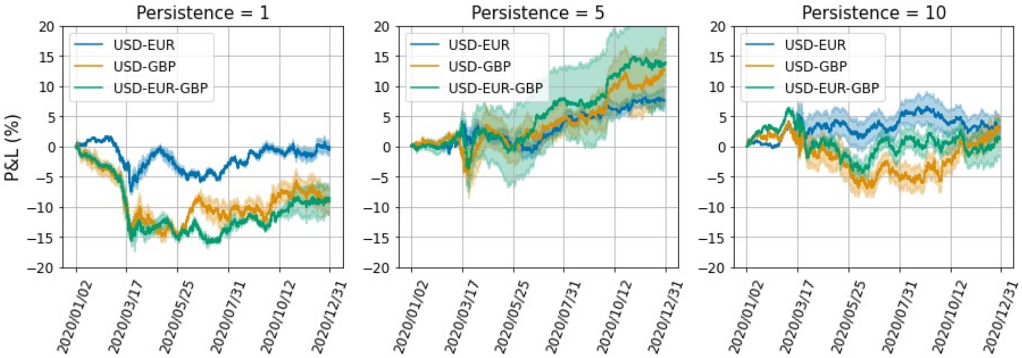
- Window of 60 price returns
- Time-step with 1-minute, 5-minute, 10-minute frequency (Persistence)

Validation: Model Selection



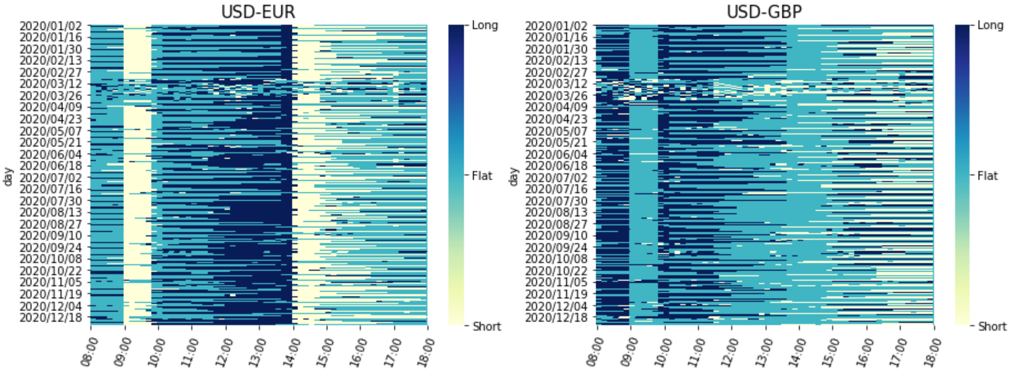
Validation on the single currency pair EURUSD, averaged over 2 seeds

Test Performances: P&L

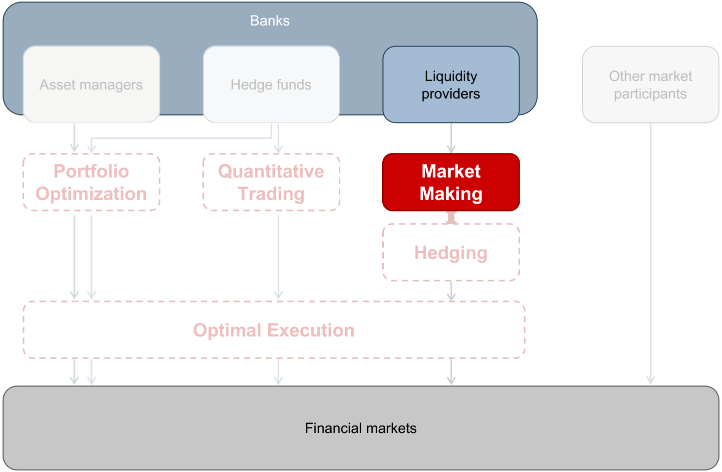


	Persistence	1	5	10
Sharpe	EUR	-0.22	1.34	0.27
Ratio	GBP	-1.37	1.93	0.63
	Both	-1.45	**2.02**	0.33

Test Performances: Heat Maps



D. Market Making in Dealer Markets



Bond Market Making

***Market making**: a sequential decision process in which at each round $t \in \{1, \ldots, T\}$ the dealer updates her bid and ask prices to maximize P&L while minimizing inventory*

PCS	Firm Name	Bid Px / Ask Px	Bid Yld / Ask Yld	BSz... x AS...	Time ↓
	Total Axe Size			205 x	
CBBT	FIT COMPOSITE	91.844 / 91.868	1.833 / 1.830	x	11:59
BVAL	BVAL (Score: 10)	91.624 / 91.640	1.858 / 1.856	x	09:00
	Last Trade	91.856	--	7.7	11:34
NOMX	NOMURA INTL PLC LDN	91.848 / 91.882	1.832 / 1.828	50 x 10	11:59
MZHO	MIZUHO INTL	91.8400 / 91.8928	1.832 / 1.827	5 x 10	11:59
IMIG	INTESA SANPAOLO IMIG	91.795 / 91.895	1.838 / 1.827	10 x 10	11:59
MSEG	MORGAN STANLEY LOND	91.847 / 91.922	1.832 / 1.823	3 x 10	11:59
BSGB	SANTANDER Ex	91.848 / 91.918	1.831 / 1.824	25 x 5	11:59
HVGO	UniCredit Bank AG	91.800 / 91.919	1.837 / 1.824	5 x 5	11:59
DZBK	DZ BANK	91.796 / 91.916	1.838 / 1.824	5 x 5	11:59
HELA	HELABA AUTO EX	91.781 / 91.930	1.840 / 1.823	5 x 5	11:59
DEKA	DEKABANK	91.806 / 91.906	1.837 / 1.825	2.5 x 2.5	11:59
BPEG	BNP PARIBAS EURO G...	91.863 / 91.937	1.830 / 1.822	2 x 2	11:59

Market Making as an MDP

State:

- price of the asset: P_t (exogenous)
- the inventory: $z_t = z_{t-1} + v_t \mathbb{I}\{won_t\}$

Actions:

- $a_1 : P^i_{t,buy}(v) = \tilde{P}_{t,buy}(v)(1 + a_1)$
- $a_2 : P^i_{t,sell}(v) = \tilde{P}_{t,sell}(v)(1 + a_2)$

Reward:

$$r_t = \underbrace{\mathbb{I}\{won_t\}|v_t(P_{t,buy/sell}(v_t) - P_t)|}_{\text{spread P\&L}} + \underbrace{z_{t-1}(P_t - P_{t-1})}_{\text{inventory P\&L}} - \underbrace{\phi(z_t)}_{\text{inventory penalty}}$$

where v_t is the size of the trade, $P_{t,buy/sell}(v_t)$ is the quote published by the market maker, z_t is the inventory, $\phi : \mathbb{R} \to \mathbb{R}^+$ is the penalty of owning a net inventory

Approaches to Market Making

Background

- **Classical approach**
  [Avellaneda and Stoikov, 2008]
  - Model the mid-price process and RFQ arrival process
  - Define the market maker's utility function
  - Model auctions as stochastic processes
- **Reinforcement Learning approach**
  [Ganesh et al., 2019]
  - Model the mid-price process and RFQ arrival process
  - Define the behavior of the other dealers

Approaches to Market Making

Background

- **Classical approach** [Avellaneda and Stoikov, 2008]
  - Model the mid-price process and RFQ arrival process
  - Define the market maker's utility function
  - Model auctions as stochastic processes
- **Reinforcement Learning approach** [Ganesh et al., 2019]
  - Model the mid-price process and RFQ arrival process
  - Define the behavior of the other dealers

Main contributions

- Model as an N-player stochastic game, with multiple competing market makers
- Solve by using mean field games and FQI

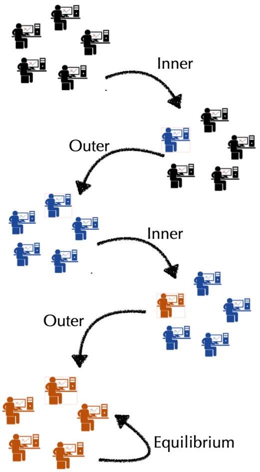
Learning in Mean-Field Games

- Assume homogeneity/anonymity
- Mean-field $\mathcal{L} \in \Delta(\mathcal{A} \times \mathcal{S})$ represents players' distribution
- Nash Equilibrium is a pair $(\pi^*, \mathcal{L}^*)$ s.t.
  $V(\pi^*, \mathcal{L}^*) \geq V(\pi, \mathcal{L}^*)$, $\forall \pi$

Algorithm 3 Model Free MFG [Guo et al., 2019]

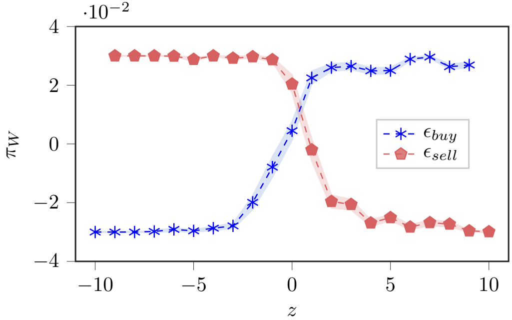
Require: mean-field $\mathcal{L}_0$, simulator $\mathcal{E}(.,.;\mathcal{L})$, iterations K

1: **for** $k \in [K]$ **do**
2: Find the single-agent optimal policy π_k with fixed $\mathcal{L}_k$
3: Update $\mathcal{L}_{k+1}$ using $\mathcal{E}(.,.;\mathcal{L})$
4: **end for**
5: **return** $(\pi_k, \mathcal{L}_k)$

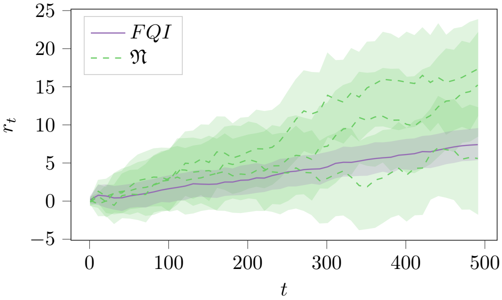


Experimental Results

Learned Policy



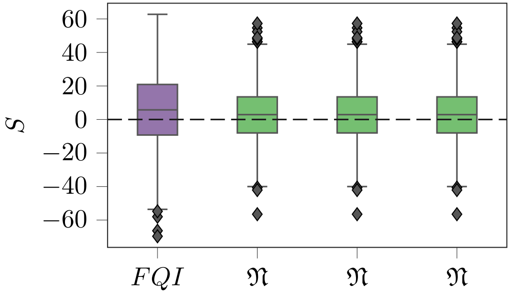
Mean dollar reward



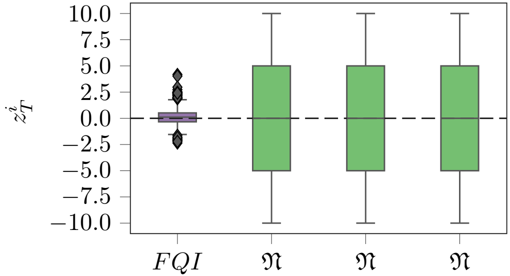
- π_W learned policy
- z: inventory
- $\mathcal{A} = \{-0.03, -0.02, ..., 0.03\}$

- ρ_t : mean dollar reward ($\phi = 0$)
- *FQI*: trained with MFG-FQI
- $\mathfrak{N}$: plays $(a_1, a_2) \sim \mathcal{N}(0, 1)$

Sharpe ratio box plot

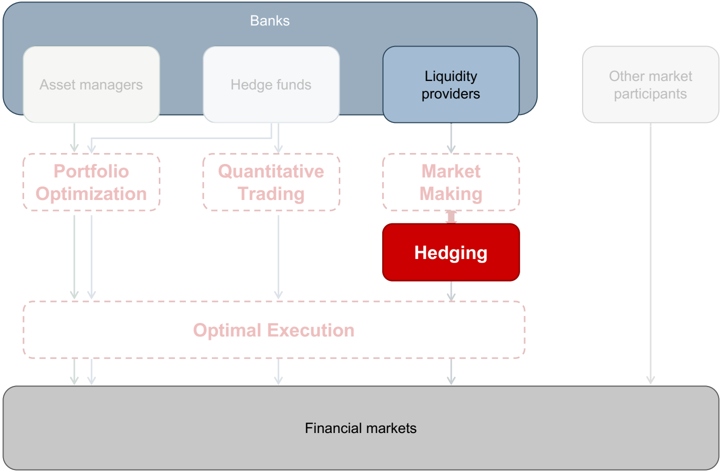


Inventory box plot



- $R = \sum_{t \leq T} \frac{\rho_t}{T}$
- Sharpe ratio $S = R/std(R)$

E. Credit Index Option Hedging with RL



Credit Index Option Hedging

A **Credit Default Swap** (CDS) is a financial derivative that allows an investor to swap or offset her credit risk with that of another investor

A **receiver** option gives the buyer the possibility of selling protection on the index at the expiry date at a spread equal to the strike

A **payer** option gives the buyer the choice of buying protection at the expiry date at a spread equal to the strike

Approaches to Option Hedging

Background

- **Classical approach**
  [Black and Scholes, 1973]
  - Model the market as GBM
  - Assume continuous time hedging
  - Assume no market frictions
  - Solve resulting PDE
- **Reinforcement Learning approach**
  [Kolm and Ritter, 2019]
  - Collect/simulate data
  - Learn to hedge

Approaches to Option Hedging

Background

- Classical approach
  [Black and Scholes, 1973]
  - Model the market as GBM
  - Assume continuous time hedging
  - Assume no market frictions
  - Solve resulting PDE
- Reinforcement Learning approach
  [Kolm and Ritter, 2019]
  - Collect/simulate data
  - Learn to hedge

Main contributions

- Use of the risk-averse policy search RL algorithm: TRVO
- Training and testing on credit index options
- Testing on real data

Financial Environment

Long payer option

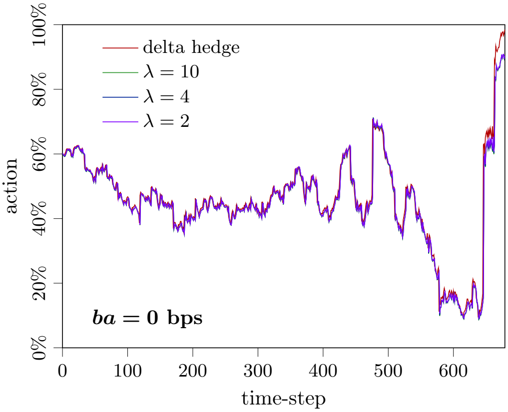
- time to maturity = 40 days
- €100mln notional
- implied volatility = 60%
- interest rates = 0
- $K(= S_0)$ = 100
- starting price (ATM) option $\sim$ €530k
- starting delta $= 0.5$

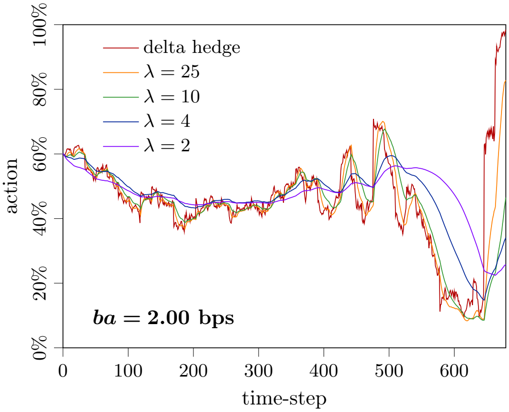
Simulated Credit Spread

- GBM
- no drift
- $\sigma = 60\%$
- $S_0 = 100$
- 17 observations per day

training on 40k episodes and testing on 2k episodes

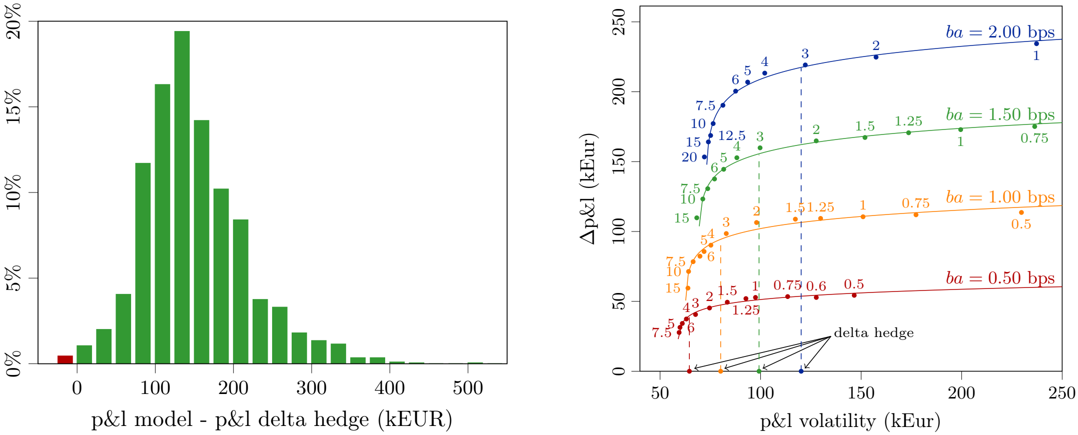
Experimental Results: with/without Transaction Costs





delta hedge with no costs $\rightarrow$ average p&l $\sim$ 0, with costs $\rightarrow$ average p&l $\sim$-€136k

Experimental Results: GBM Simulated Market



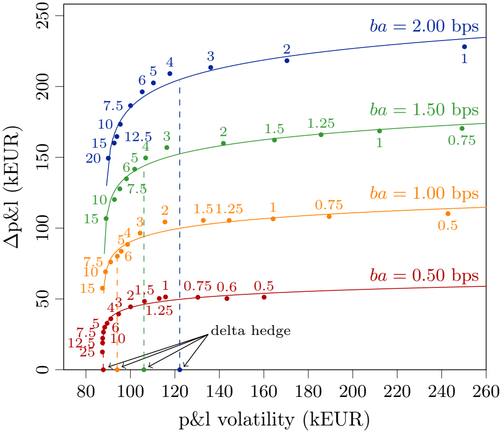
distribution of P&L of $\lambda = 4$ agent with $ba = 1.5bps$

Experimental Results: Heston Simulated Market

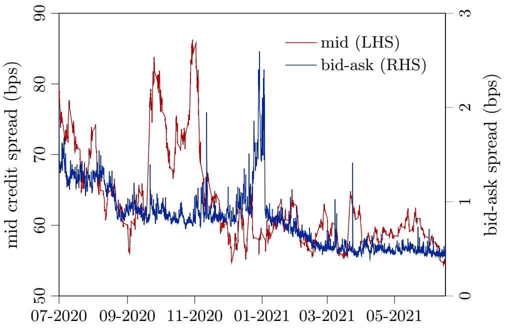
Testing on 2k heston simulated episodes

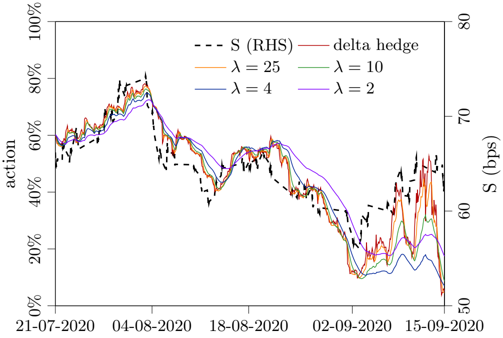
$$dS_t = \sqrt{\nu_t}\, S_t\, dW_t^S$$
$$d\nu_t = \kappa\,(\theta - \nu_t)\, dt + \xi\sqrt{\nu_t} dW_t^\nu$$

$\nu_0 = 60\%^2$, $\kappa = 2$, $\theta = \nu_0$, $\xi = 0.9$
no correlation between the stochastic terms dW_t^S and dW_t^ν.



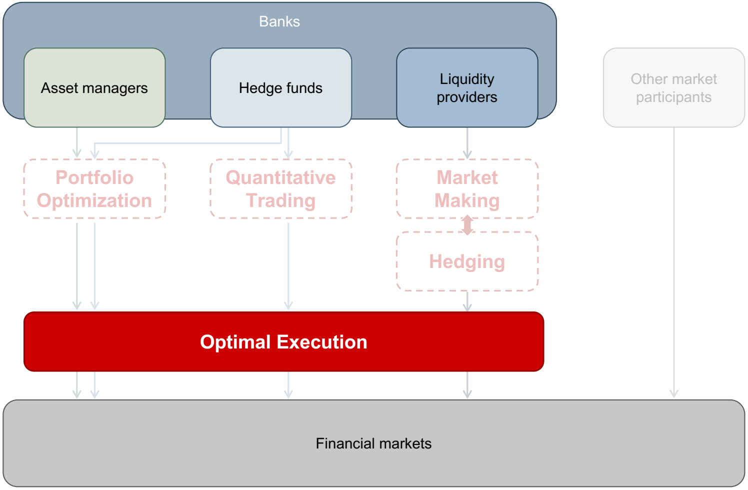
Experimental Results: Real Data





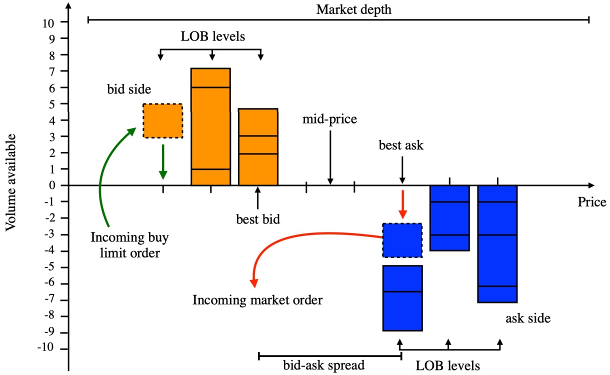
Testing on real data, with option $\sigma = 60\%$ and $ba = 1bps$

F. Optimal Execution with RL



Optimal Execution Problem

Optimal execution: *a sequential decision process in which at each round $t \in \{1, ..., T\}$ over the maximum execution time T and number of time-steps $N+1$, the trader decides what fraction of the total X shares to execute, to minimize the difference between the arrival price and the execution price*



Approaches to Optimal Execution

Background

- Practitioner approach
  - TWAP$= \frac{X}{N}\sum_{k=0}^{N} P_k$
- Classical approach
  [Almgren and Chriss, 2001]
  - Model the mid-price process
  - Model the market impact
  - Minimize expected shortfall
- Reinforcement Learning approach
  [Hendricks and Wilcox, 2014]
  - Collect/simulate data
  - Model the market impact
- Multi agent approach using ABIDES
  - Learn in a multi-agent simulation

Approaches to Optimal Execution

Background

- Practitioner approach
  - TWAP$= \frac{X}{N} \sum_{k=0}^{N} P_k$
- Classical approach [Almgren and Chriss, 2001]
  - Model the mid-price process
  - Model market impact
  - Minimize expected shortfall
- Reinforcement Learning approach [Hendricks and Wilcox, 2014]
  - Collect/simulate data
  - Model market impact
- Multi agent approach using ABIDES
  - Learn in a multi-agent simulation

Main contributions

- Use of FQI to learn multiple execution policies in a multi-agent simulation
- Use of Thompson Sampling to decide which execution policy to use

Optimal Execution as an MDP

MDP Formulation

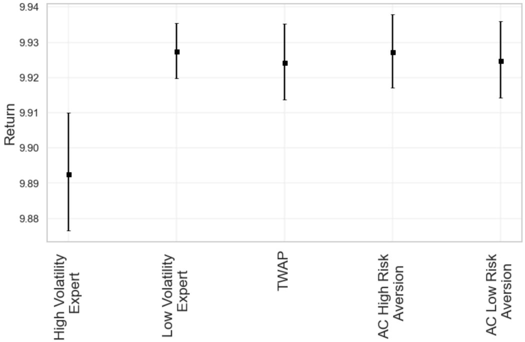
- $a_t \in \{0, 0.2, 0.4, ..., 4\}$ represents how much of TWAP *i.e.*, $\frac{X}{N}$ to execute
- $s_t =$ stylized microstructural order book facts and internal agent information
- $r_t = \left(1 - \frac{1}{P_{t_{\text{fill}}}}|P_{t_{\text{fill}}} - P_{\text{arrival}}|\right) \lambda \frac{n_t}{X}$

Environment Formulation

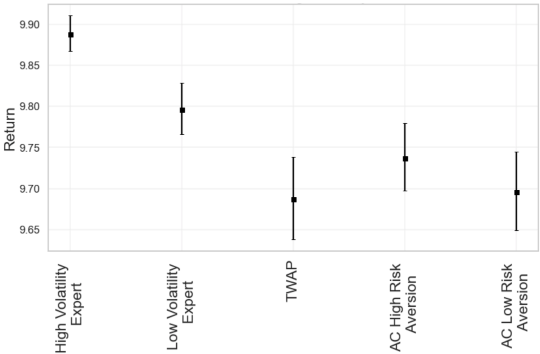
- $X = 50,000$
- $N = 180$, $T = 30$ minutes, $\tau = 10$
- Training on 2,000 executions
- Training with FQI [Ernst et al., 2005]

Experimental Performance on Two Scenarios

Performance on Low Volatility Scenario

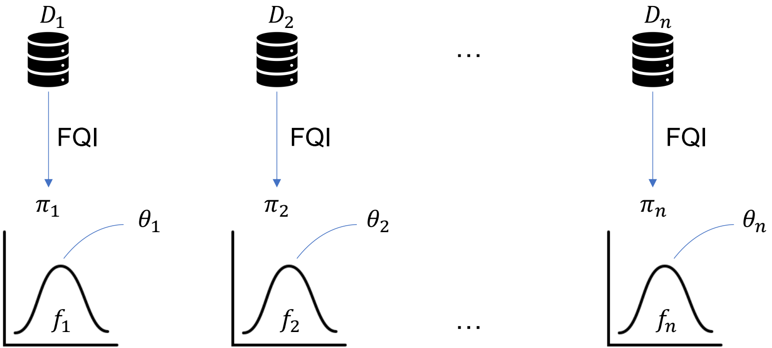


Performance on High Volatility Scenario



Average return over 50, 30-minute executions with 95% confidence intervals

Thompson Sampling for Optimal Execution



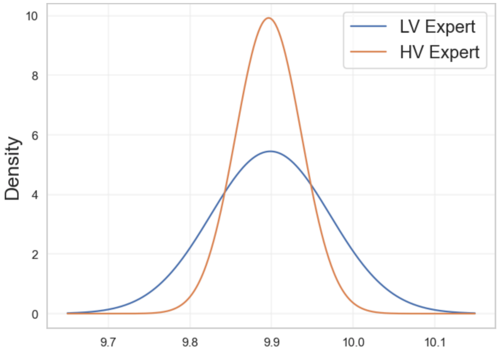
$i_t = argmax_i \theta_i$

Run π_i observe r_i and update f_i

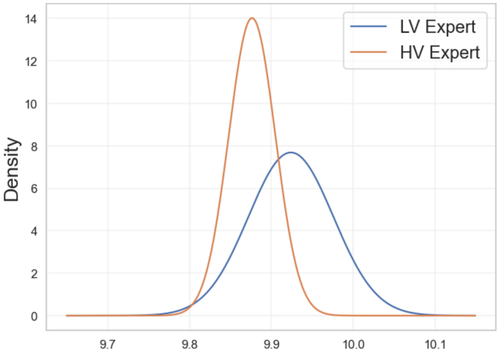
iterate

Thompson Sampling - Low Volatility Scenario

Distribution after 5 TS iterations

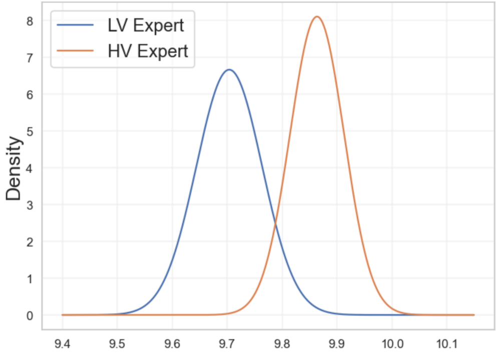


Distribution after 10 TS iterations

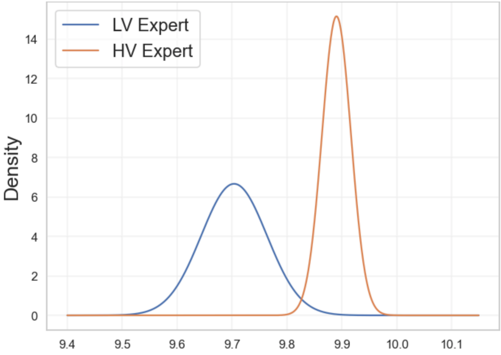


Thompson Sampling - High Volatility Scenario

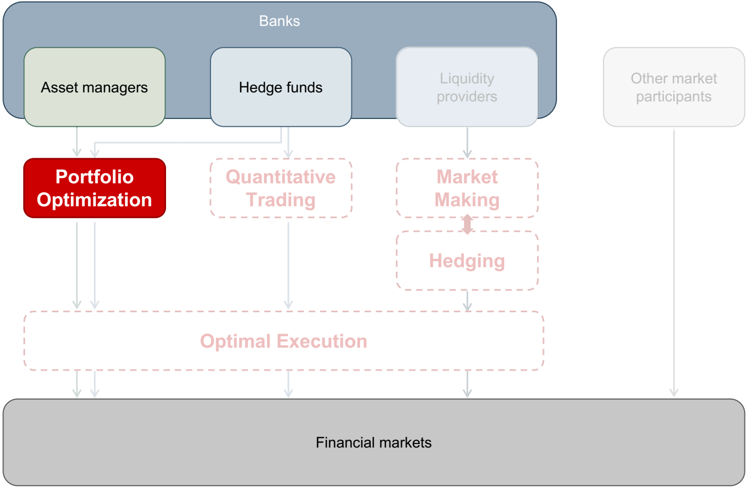
Distribution after 5 TS iterations



Distribution after 10 TS iterations



G. Conservative Online Convex Optimization



Context - Beating a Market Index

A ***market index*** *is a collection of financial assets, commonly stocks. The returns of the market index are calculated as a weighted average of the returns of the constituents.*

The objective of the asset manager is to invest in a subset of the components of the index or to use a different weighting than the index, to outperform the index itself

Conservativeness Objective

$$L_t \leq \tilde{L}_t(1+\alpha),\ \forall t$$

- $\tilde{L}_T$: cumulative loss of the default parameter $\tilde{\theta} \in \Theta$
- $\alpha > 0$: conservativeness level

Conservativeness Objective in OPO

$$W_t(\mathfrak{U}) \geq \tilde{W}_t(1-\kappa),\ \forall t$$

Approaches to Portfolio Optimization

Background

- **Modern Portfolio Optimization**
  [Markowitz, 1952]
  - Calculate historical variance and correlations
  - Single period
- **Intertemporal CAPM**
  [Merton, 1969]
  - Make assumptions on asset dynamics
  - Multi period
- **Online Portfolio Optimization**
  [Cover and Ordentlich, 1996]
  - Adversarial market
  - From expert learning field

Approaches to Portfolio Optimization

Background

- Modern Portfolio Optimization
  [Markowitz, 1952]
  - Calculate historical variance and correlations
  - Single period
- Intertemporal CAPM
  [Merton, 1969]
  - Make assumptions on asset dynamics
  - Multi period
- Online Portfolio Optimization
  [Cover and Ordentlich, 1996]
  - Adversarial market
  - From expert learning field

Main contributions

Conservative online convex optimization
[Bernasconi de Luca et al., 2021]

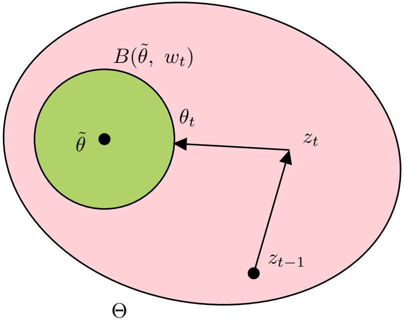
- Beating a benchmark in OPO

The Conservative Projection Algorithm [Bernasconi de Luca et al., 2021]

Algorithm 4 CP-$\mathcal{A}$

Require: Algorithm $\mathcal{A}$, $\alpha > 0$, $\tilde{\theta} \in \Theta$

1: Set $\tilde{L}_0 \leftarrow 0$, $L_0 \leftarrow 0$, and $\beta_0 \leftarrow 1$
2: **for** $t \in [T]$ **do**
3: Get point $z_t \leftarrow \mathcal{A}(g_1, \dots, g_{t-1})$
4: Compute $\omega_t := \left[1 - \left(\frac{L_{t-1} - (1+\alpha)\tilde{L}_{t-1} - \alpha\varepsilon_l}{DG} + 1\right)^+\right] D$
5: Select $\theta_t = \Pi_{B(\tilde{\theta}, \omega_t)}(z_t)$
6: Suffer loss $f_t(\theta_t)$
7: Observe $f_t(z_t)$ and $f_t(\tilde{\theta})$
8: Set $g_t(z_t) \leftarrow (1 - \beta_t) f_t(z_t)$ with $\beta_t = \begin{cases} 1 - \frac{\omega_t}{||z_t - \tilde{\theta}||_2} & z_t \notin B(\tilde{\theta}, \omega_t) \\ 0 & z_t \in B(\tilde{\theta}, \omega_t) \end{cases}$
9: **end for**



Main Theoretical Result

Theorem

For any Online Convex Optimization algorithm $\mathcal{A}$, with regret $R_T(\mathcal{A})$ and $\alpha > 0$, CP-$\mathcal{A}$ attains regret:

$$R_T(\text{CP-}\mathcal{A}) \leq R_T(\mathcal{A}) + \tau DG$$

where $\tau = \mathcal{O}(\alpha^{-1})$. Moreover CP-$\mathcal{A}$ is a conservative algorithm

$D := \sup_{x,y \in \Theta} ||x - y||_2$ is a bound on the diameter of the parameter space Θ

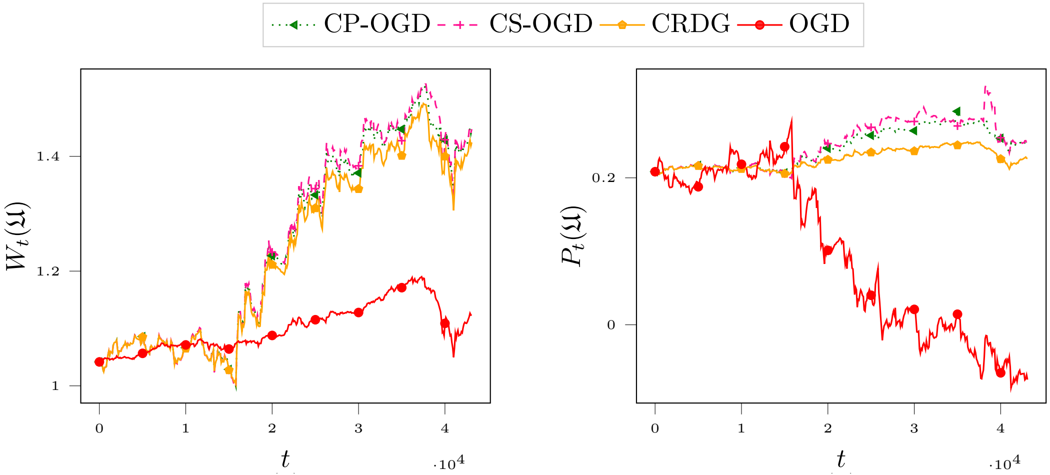
$G := \sup_{x \in \Theta} ||\nabla f_t(x)||_2$ is the upper bound on the norm of the gradient of the loss $f_t(\cdot)$

Experimental Setup

Dataset with minute prices of S&P component stocks from 09/2017 to 02/2018
$\tilde{\theta} = 100$ randomly chosen stocks

- Metrics
  - Wealth: $W_T(\mathfrak{U}) = \prod_{t=1}^{T} \langle \mathbf{a}_t, \mathbf{y}_t \rangle$
  - Wealth budget: $P_t(\mathfrak{U}) = W_t(\mathfrak{U}) - (1-\kappa)\tilde{W}_t$
- Algorithms
  - Online Gradient Descent [Zinkevich, 2003]
  - CRDG [Streeter and McMahan, 2012]
  - CS-OGD
  - **CP-OGD**

Experimental Results



H. State of the Art

Option Hedging with RL

- Cannelli, L., Nuti, G., Sala, M., and Szehr, O. (2020). **Hedging using reinforcement learning: Contextual k-armed bandit versus q-learning.**

  arXiv preprint

- Kolm, P. N. and Ritter, G. (2019). **Dynamic replication and hedging: A reinforcement learning approach.**

  The Journal of Financial Data Science, 1(1):159–171

- Buehler, H., Gonon, L., Teichmann, J., and Wood, B. (2019). **Deep hedging.**

  Quantitative Finance, pages 1–21

- Halperin, I. (2019). **The qlbs q-learner goes nuqlear: fitted q iteration, inverse rl, and option portfolios.**

  Quantitative Finance, pages 1–11

- Cao, J., Chen, J., Hull, J. C., and Poulos, Z. (2019). **Deep hedging of derivatives using reinforcement learning.**

  Available at SSRN

Trading with RL

- Théate, T. and Ernst, D. (2021). **An application of deep reinforcement learning to algorithmic trading.**

  Expert Systems with Applications, 173:114632
- Briola, A., Turiel, J., Marcaccioli, R., and Aste, T. (2021). **Deep reinforcement learning for active high frequency trading.**

  arXiv preprint
- Kolm, P. N. and Ritter, G. (2020). **Modern perspectives on reinforcement learning in finance.**

  Available at SSRN
- Liu, X.-Y., Yang, H., Chen, Q., Zhang, R., Yang, L., Xiao, B., and Wang, C. D. (2020). **Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance.**

  arXiv preprint
- Meng, T. L. and Khushi, M. (2019). **Reinforcement learning in financial markets.**

  Data, 4(3):110

Market Making with RL

- Spooner, T. and Savani, R. (2020). **Robust market making via adversarial reinforcement learning.**

  IJCAI

- Ganesh, S., Vadori, N., Xu, M., Zheng, H., Reddy, P., and Veloso, M. (2019). **Reinforcement learning for market making in a multi-agent dealer market.**

  arXiv preprint

- Guéant, O. and Manziuk, I. (2019). **Deep reinforcement learning for market making in corporate bonds: beating the curse of dimensionality.**

  Applied Mathematical Finance, 26(5):387–452

- Spooner, T., Fearnley, J., Savani, R., and Koukorinis, A. (2018). **Market making via reinforcement learning.**

  arXiv preprint

Optimal Execution with RL

- Karpe, M., Fang, J., Ma, Z., and Wang, C. (2020). **Multi-agent reinforcement learning in a realistic limit order book market simulation.**

  arXiv preprint

- Lin, S. and Beling, P. A. (2020). **An end-to-end optimal trade execution framework based on proximal policy optimization.**

  In *IJCAI*, pages 4548–4554

- Ning, B., Lin, F. H. T., and Jaimungal, S. (2018). **Double deep q-learning for optimal execution.**

  arXiv preprint

- Hendricks, D. and Wilcox, D. (2014). **A reinforcement learning extension to the almgren-chriss framework for optimal trade execution.**

  In *CIFEr*, pages 457–464. IEEE

- Nevmyvaka, Y., Feng, Y., and Kearns, M. (2006). **Reinforcement learning for optimized trade execution.**

  In *ICML*, pages 673–680

Online Portfolio Optimization

- Hoi, S. C., Sahoo, D., Lu, J., and Zhao, P. (2021). **Online learning: A comprehensive survey.**

  Neurocomputing, 459:249–289

- Li, B., Wang, J., Huang, D., and Hoi, S. (2018). **Transaction cost optimization for online portfolio selection.**

  QUANT FINANC, 18(8):1411–1424

- Li, B. and Hoi, S. (2014). **Online portfolio selection: A survey.**

  ACM COMPUT SURV, 46(3):35

- Li, B., Zhao, P., Hoi, S., and Gopalkrishnan, V. (2012). **Pamr: Passive aggressive mean reversion strategy for portfolio selection.**

  MACH LEARN, 87(2):221–258

- Hazan, E., Agarwal, A., and Kale, S. (2007). **Logarithmic regret algorithms for online convex optimization.**

  MACH LEARN, 69:169–192

Multiperiod Trading/Portfolio Optimization

- Kolm, P. N. and Ritter, G. (2020). **Modern perspectives on reinforcement learning in finance.**

  Available at SSRN

- Ritter, G. (2017). **Machine learning for trading.**

  Available at SSRN

- Boyd, S., Busseti, E., Diamond, S., Kahn, R. N., Koh, K., Nystrup, P., and Speth, J. (2017).

  Multi-period trading via convex optimization.

  arXiv preprint

- Gârleanu, N. and Pedersen, L. H. (2013). **Dynamic trading with predictable returns and transaction costs.**

  The Journal of Finance, 68(6):2309–2340

- Merton, R. C. (1969). **Lifetime portfolio selection under uncertainty: The continuous-time case.**

  The review of Economics and Statistics, pages 247–257

My Publications I

- Vittori, E., Likmeta, A., and Restelli, M. (2021). **Monte carlo tree search for trading and hedging.**

  In *ICAIF*
- Riva, A., Bisi, L., Sabbioni, L., Liotet, P., Vittori, E., Trapletti, M., Pinciroli, M., and Restelli, M. (2021). **Learning fx trading strategies with fqi and persistant actions.**

  In *ICAIF*
- Bernasconi de Luca, M., Vittori, E., Trovò, F., and Restelli, M. (2021). **Conservative online convex optimization.**

  In *ECML*

My Publications II

- Vittori, E., Bernasconi de Luca, M., Trovò, F., and Restelli, M. (2020a). **Dealing with transaction costs in portfolio optimization: Online gradient descent with momentum.**

  In *ICAIF*

- Vittori, E., Trapletti, M., and Restelli, M. (2020b). **Option hedging with risk averse reinforcement learning.**

  In *ICAIF*

- Bisi, L., Sabbioni, L., Vittori, E., Papini, M., and Restelli, M. (2020). **Risk-averse trust region optimization for reward-volatility reduction.**

  In *IJCAI. Special Track on AI in FinTech.*

Research Venues

Machine Learning

- Neurips
- ICML
- IJCAI
- AAAI
- ECML
- Journal of Machine Learning Research

ML in Finance

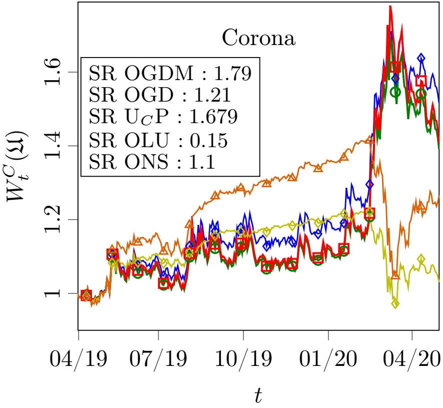
- ICAIF
- The Journal of Financial Data Science

Quant Finance

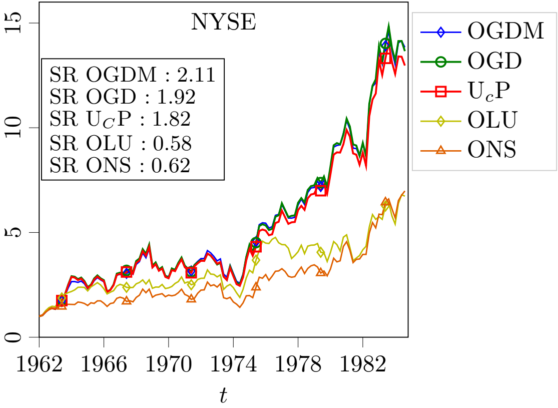
- Mathematical Finance
- Finance and Stochastics
- Applied Mathematical Finance
- Risk Magazine
- Journal of Empirical Finance
- Journal of Computational Finance

I. Additional Material

Experiments: Wealth $W_T^C(\mathfrak{U})$

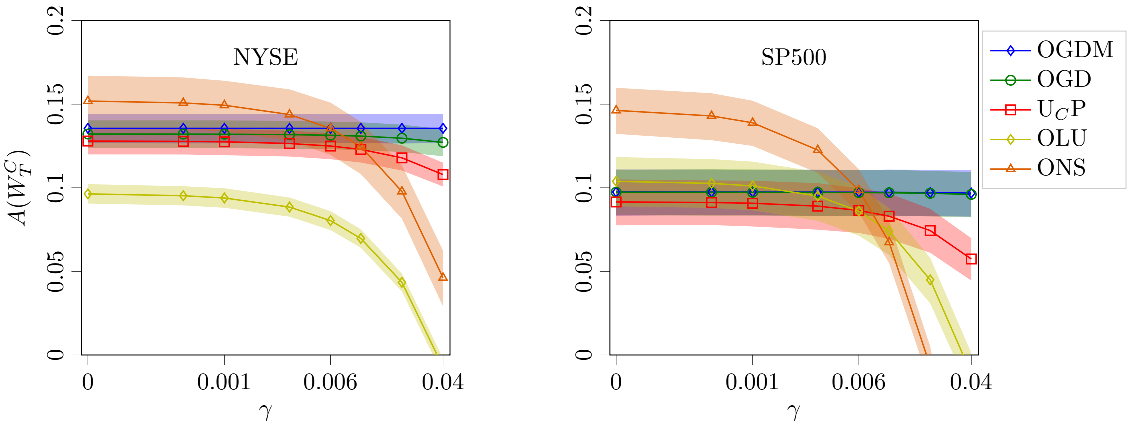


Specific run, on the Corona dataset for $\gamma = 0$



Specific run on 5 stocks of the NYSE(O) for $\gamma = 0.01$

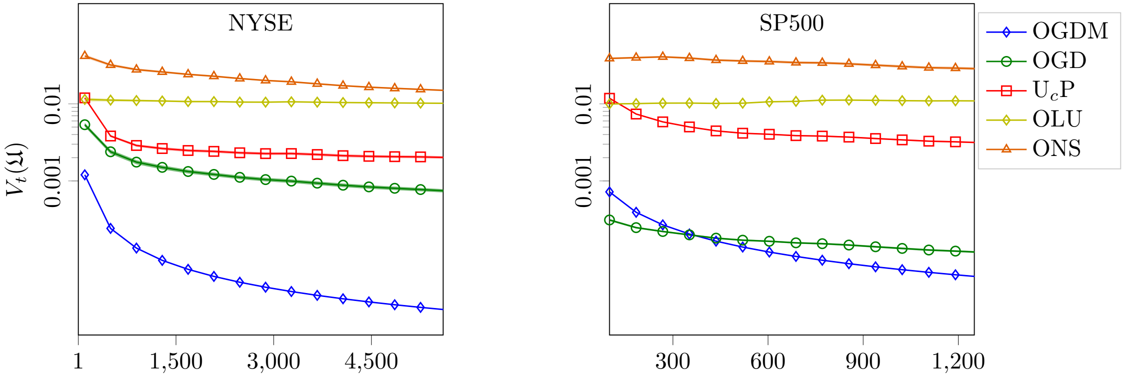
Experiments: Average APY



Average Average Annual Percentage Yield $A(W_T)$ computed on the wealth $W_T^C(\mathbf{a}_{1:T}, \mathbf{y}_{1:T})$:

$$A(W_T) = W_T^{250/T} - 1$$

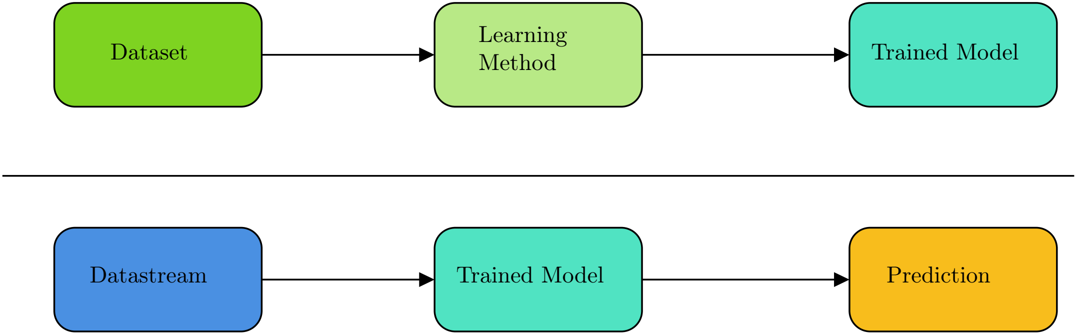
Experiments: Average variation of the portfolio



Average variation of the portfolio incurred on a varying time horizon t:

$$V_t(\mathfrak{U}) = \frac{C_t(\mathfrak{U})}{\gamma t}$$

Problem Context



Empirical Risk Minimization vs Online Optimization

ERM

- Samples are generated from a distribuition
- Minimize expected loss given a colletion of samples (dataset)
- Subject to Adversarial attacks and Concept Drift
- Voice to text, image classification, Natural Language Processing

Online Optimization [Hazan, 2019]

- Allows samples to be generated by an adversary
- No assumption on the distribuition of the data
- No guarantees on the first phase of the learning process
- Spam classification, Malware detection, Fraud detection

How to obtain a best of both worlds approach and obtain an online algorithm which has controlled performance at each time?

Algorithm 5 CS-$\mathcal{A}$

Require: Online learning algorithm $\mathcal{A}$, conservativeness level $\alpha > 0$, default parameter $\tilde{\theta} \in \Theta$

```
Set L̃_0 ← 0, L_0 ← 0
for t ∈ [T] do
    if L_{t−1} + ε_u − (1 + α)ε_l ≤ L̃_{t−1}(1 + α) then
        z_t ← A(f_{t−1}(z_{t−1}))
        Select θ_t ← z_t
    else
        z_t ← z_{t−1}
        Select θ_t ← θ̃
    end if
    Suffer loss f_t(θ_t)
    Observe feedback f_t(z_t) and f_t(θ̃)
end for
```

Experimental Setup

Tasks

- Linear Regression: Synthetic data
- Binary Classification: IMDB and SpamBase

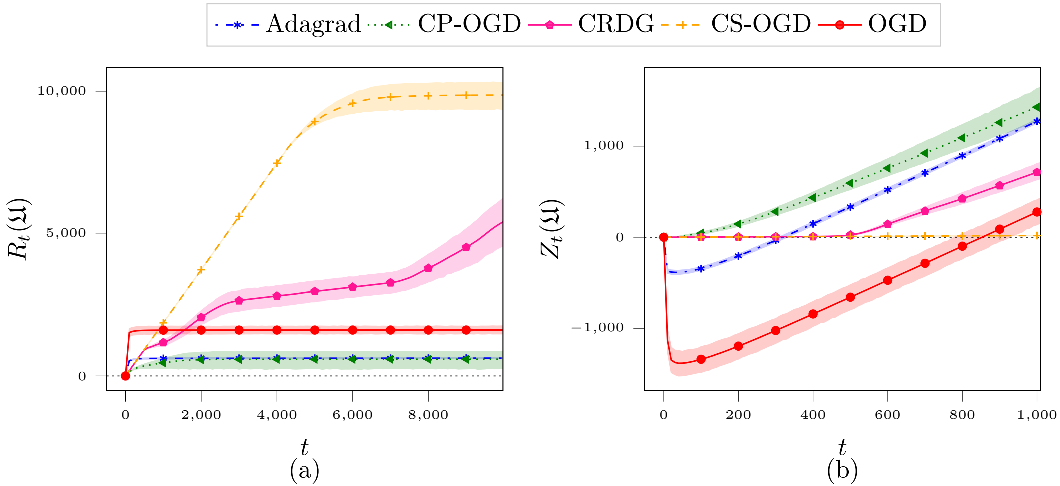
Metrics

- Budget: $Z_t = \tilde{L}_t(1 + \alpha) - L_t$
- Regret: R_t

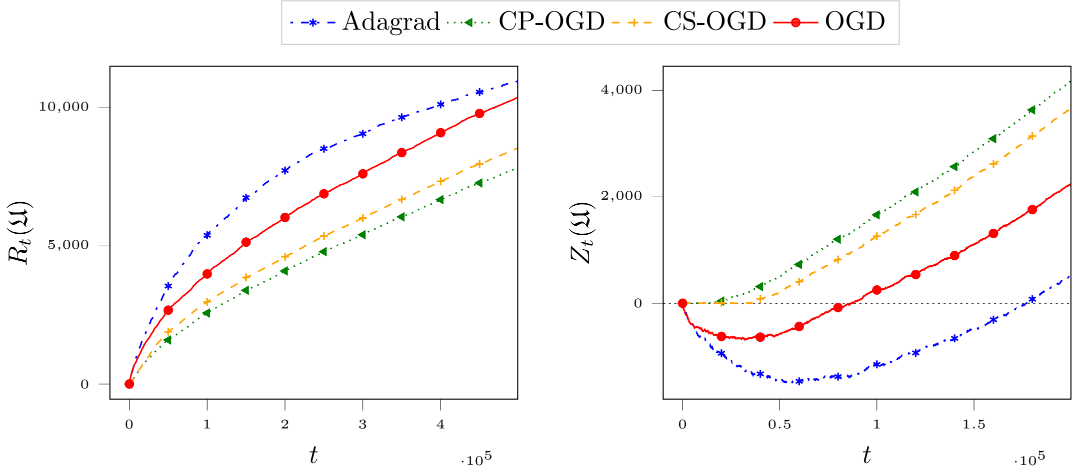
Algorithms

- Online Gradient Descent [Zinkevich, 2003]
- ADAGRAD [Duchi et al., 2011]
- CRDG [Streeter and McMahan, 2012]
- CS-OGD
- **CP-OGD**

Results: Synthetic Data



Reuslts: IMDB



(a)

(b)

ABIDES [Byrd et al., 2019]

ABIDES realistically replicates the financial market environment reproducing the characteristics of electronic markets:

- Continuous double-auction trading
- Network latency and agent computation delays
- Communication solely by means of standardized message protocols

It is possible to create a multi-agent composition using pre-defined agents such as the **exchange** agent, **value** agents, **momentum** agents, **noise** agents and **market maker** agents or using **custom made** agents

The price process is described by a **fundamental value**

ABIDES Multi-agent Market Simulator

ABIDES [Byrd et al., 2019] reproduces the characteristics of electronic markets such as continuous double-auction trading and network latency.