Risk-averse Reinforcement Learning for Algorithmic Trading

Yun Shen\(^1\)  \hspace{0.5cm} \text{Ruihong Huang}\(^2,3\)  \hspace{0.5cm} \text{Chang Yan}\(^2\)  \hspace{0.5cm} \text{Klaus Obermayer}\(^1\)

\(^1\text{Technische Universität Berlin}\)
\(^2\text{Humboldt-Universität zu Berlin}\)
\(^3\text{LOBSTER Team}\)

IEEE CIFEr, London
March 28, 2014
Introduction

- Transaction cost:

\[ TC = X(P_d - P_0) + \sum x_i P_j - \sum x_i P_0 + (X - \sum x_i)(P_n - P_0) + \text{visible} \]

  - visible: commission fees, taxes, etc.

- Task: to liquid a huge inventory over a short time horizon.

- Data: high-frequency (millisecond) limit orders in NASDAQ

- Method: reinforcement learning (RL) + risk control

Risk

- Uncertain future prices, volumes, etc.
  - Example: 2010 flash crash
- Standard RL is risk-neutral
  - does not take risk into consideration
  - perform badly when outlier events happen
- Variance as a risk measure:
  - non-Gaussian noise is computationally infeasible
  - Gaussian noise is not always the case

⇒ need new measures of risk and new algorithms

Nevmyvaka et al., ICML, 2006; Almgren & Chriss, J. of Risk, 2001

---

"On May 6, 2010, the prices of many U.S.-based equity products experienced an extraordinarily rapid decline and recovery. That afternoon, major equity indices in both the futures and securities markets, each already down over 4% from their prior-day close, suddenly plummeted a further 5-6% in a matter of minutes before rebounding almost as quickly." – CFTC & SEC Report
Markov decision processes

Objective: \( \sum x_j P_j - \sum x_j P_0 + (X - \sum x_j)(P_n - P_0) \)

\[
\max_{\pi} J(\pi, s) := \mathbb{E} [S_T | S_1 = s, \pi] = \mathbb{E} \left[ \sum_{t=1}^{T} R(S_t, A_t) | S_1 = s, \pi \right]
\]

- reward function \( R : S \times A \times \Omega \rightarrow \mathbb{R} \)
- policy \( \pi = [\pi_1, \pi_2, \ldots, \pi_T], \pi_t : S \rightarrow A \)
- key assumption: Markov – \( \mathbb{P}(S_{t+1} | \mathcal{F}_t) = P(S_{t+1} | S_t, A_t) \)

\[
J(\pi, s) = \mathbb{E}_{S_1 = s}^{\pi_1} \left[ R(S_1, A_1) + \mathbb{E}_{S_2}^{\pi_2} \left[ R(S_2, A_2) + \ldots + \mathbb{E}_{S_T}^{\pi_T} [R(S_T, A_T)] \ldots \right] \right]
\]

- Adding risk:

\[
\max_{\pi} \mathbb{E}^{\pi} [S_T] - \lambda V^{\pi} [S_T]
\]

- \( \lambda \) controls the risk sensitivity.
- example of \( V \): standard deviation
- it is difficult\(^3\) to solve the problem except the case with Gaussian noise

---

see, e.g., Puterman, 1994; Almgren & Chriss, 2000; Mannor & Tsitsiklis, 2010.
Evaluation function/risk measure

\[ \mathbb{E}_{S_1=s}^{\pi_1} \left[ R(S_1, A_1) + \mathbb{E}_{S_2}^{\pi_2} \left[ R(S_2, A_2) + \ldots + \mathbb{E}_{S_T}^{\pi_T} [R(S_T, A_T)] \right] \right] \Rightarrow \]

\[ U_{S_1=s}^{\pi_1} \left[ R(S_1, A_1) + U_{S_2}^{\pi_2} \left[ R(S_2, A_2) + \ldots + U_{S_T}^{\pi_T} [R(S_T, A_T)] \right] \right] \]

\[ U(\cdot|s, a) \text{ is a risk measure for all } (s, a) \]

- monotonicity
- translation invariance
- concavity/coherency

Utility-based shortfall\(^2\):

\[ U_{s,a}(X) = \sup \{ m \in \mathbb{R} | \mathbb{E}_{s,a} [u(X - m)] \geq 0 \} \]

- \( u \): a concave, continuous and strictly increasing function satisfying \( u(0) = 0 \).
- concave \( u \Leftrightarrow \) risk averse.

---

Utility-based shortfall

- \( U_{s,a}(X) = \sup \{ m \in \mathbb{R} \mid \mathbb{E}_{s,a}[u(X - m)] \geq 0 \} \)
- we consider

\[
u(x) = \begin{cases} \frac{1}{\lambda}[(x + 1)^\lambda - 1] & x \geq 0 \\ \frac{x}{\lambda} & x < 0 \end{cases}
\]

- \( \lambda \in (0, 1) \) controls the degree of risk-averseness.
- \( \lambda = 1: U_{s,a}(\cdot) = \mathbb{E}_{s,a}(\cdot). \)
- Example: \( \{(r_1, p), (r_2, 1 - p)\} (r_1 > r_2) \)
- subjective probability: \( w(p) = \frac{u(p) - r_2}{r_1 - r_2} \) (prospect theory)
Risk-averse reinforcement learning

- at tth time point, repeat a at state s N times ⇒ \( \{R_i, s'_i\}_{i=1,2,...,N} \).
- iterative update:

\[
Q^{(i+1)}_t(s, a) = Q^{(i)}_t(s, a) + \frac{1}{i} u \left( R_i + \max_a Q^{(i+1)}_t(s'_i, a) - Q^{(i)}_t(s, a) \right) \quad (*)
\]

- \( \pi^{(N)}_t(s) = \arg \max_{a \in A} Q^{(N)}_t(s, a) \rightarrow \pi^*_t \), the optimal policy, as \( N \rightarrow \infty \)

---


---

\[
\text{initialize } Q_{T+1}(s, a) = 0 \text{ for all } s \in S, a \in A; \\
\text{for } t = T \text{ to } 1 \text{ do} \\
\hspace{1em} \text{initialize } Q_t(s, a) = 0 \text{ for all } s \in S, a \in A; \\
\hspace{1em} \text{for each state } s \in S \text{ and } a \in A \text{ do} \\
\hspace{2em} \text{for } n = 1 \text{ to } N \text{ do} \\
\hspace{3em} \text{execute action } a \text{ at } s \text{ to obtain sampled reward } R \text{ and successive state } s'; \\
\hspace{3em} \text{update } Q_t(s, a) \text{ according to (*)}; \\
\hspace{2em} \text{end for} \\
\hspace{1em} \text{end for} \\
\text{end for}
\]

- needs no knowledge of transition model \( P(S_{t+1}|S_t, A_t) \) – data driven;
- online alg.: adapts to new data easily;
- parallel computing is possible;
- risk control by \( u \), specifically, \( \lambda \).
Problem Setting

Task: sell $V$ shares of AMZN in NASDAQ within $H$ min.

Data
- Provided by LOBSTER† with two price levels, i.e., only two best asks and bids.
  - Training: 01.05.2009 – 30.04.2010
  - Test: 01.05.2010 – 31.10.2010
- Test period contains the flash crash on 06.05.2010.

† [http://lobsterdata.com](http://lobsterdata.com)

Performance evaluation:
- $\text{cost} = \frac{\text{mid-quote at time 0} - \text{average execution price}}{\text{mid-quote at time 0}} \times 10^4$
- Risk
  - Standard deviation of costs
  - 95%-quantile cost
MDP formulation

- **time resolution**
  - total time horizon $H = 10\text{min.}$
  - Limit orders are submitted to the market at $t = n \cdot \frac{H}{T}, n = 0, 1, \ldots, T - 1$.

- **states**
  - $V$ = the target volume, $I$ = the number of inventory units.
  - state $i = \lfloor v \cdot I/V \rfloor$.
  - market variables: spread, vol. misbalance, signed vol. etc.

- **actions**: $a = \text{submit a sell order at price } \text{ask} - a$ (unit: US cent) with all remaining shares.

- **rewards**: the cash inflow resulted from a (partial) execution of the limit order placed at $\text{ask} - a$.

---

Results I: Tuning of $\lambda$

- **Average trading cost**
- **Standard deviation**
- **95% quantile cost**
- **Cost at flash crash**

Symbols:
- $\lambda$ values from 0.1 to 1
- V=20k, T=5
- V=20k, T=10

Legend:
- Triangle: risk neutral
Results II: Flash Crash

AMZN May 6, 2010

mid quote price ($) vs. time

V=20000, T=10, I=10

trading cost (unit: basis point) vs. time

risk neutral (RN) vs. risk averse (RA)
Results II: Flash Crash

Trading costs on the flash crash spot.
Results III: Overall Performance

- Average trading cost
- Standard deviation
- 95% quantile cost

Graphs showing performance metrics for different scenarios and configurations.
Conclusion and outlook

- **Conclusion:** our novel risk-averse RL
  - significantly reduces the trading cost at the spot of flash crash
  - remarkably lowers down risk in the whole test period at the price of a slight increase of average trading cost

- **Outlook**
  - market impact
  - expand state space: test with various market variables.
  - expand action space with volume number.
  - other $u$-functions, even risk-seeking type?!