Algorithmic Trading and Computational Finance

Michael Kearns
Computer and Information Science
University of Pennsylvania

STOC Tutorial
NYC
May 19 2012

Special thanks: Yuriy Nevmyvaka, SAC Capital
Takeaways

• There are many interesting and challenging algorithmic and modeling problems in “traditional” financial markets
• Many (online) machine learning problems driven by rich & voluminous data
• Often driven by mechanism innovation & changes
• Almost every type of trading operates under reasonably precise constraints
  – high frequency trading: low latency, short holding period
  – market-making: offers on both sides, low inventory
  – optimized execution: performance tied to market data benchmarks (e.g. VWAP)
  – proprietary trading/statistical arbitrage: many risk limits (Sharpe ratio, concentration, VAR)
• These constraints provide structure
• Yield algorithm, optimization and learning problems
Financial Markets Field Guide
(“Biodiversity” of Wall Street)

• Retail traders
  – individual consumers directly trading for their own accounts (e.g. E*TRADE baby)
• “Buy” side
  – large institutional traders: portfolio managers; mutual and pension funds; endowments
  – often have precise metrics and constraints; e.g. tracking indices
  – percentage-based management fee
• “Sell” side
  – brokerages providing trading/advising/execution services
  – “program trading” → “algorithmic trading”: automated strategies for optimized execution
  – profit from commissions/fees
• Market-makers and specialists
  – risk-neutral providers of liquidity
  – (formerly) highly regulated
  – profit from the “bid-ask bounce”; averse to strong directional movement
  – automated market-making strategies in electronic markets (HFT)
• Hedge funds and proprietary trading
  – groups attempting to yield “outsized” returns on private capital (= beat the market)
  – can take short positions
  – relatively unregulated; but also have significant institutional investment
  – heavy quant consumers: “statistical arbitrage”, modeling, algorithms
  – typically take management fee and 20% of profits
• All have different goals, constraints, time horizons, technology, data, connectivity…
Outline

• I. Market Microstructure and Optimized Execution
  – online algorithms and competitive analysis
  – reinforcement learning for optimized execution
  – microstructure and market-making

• II. Mechanism Innovation: a Case Study
  – difficult trades and dark pools
  – the order dispersion problem
  – censoring, exploration, and exploitation

• III. No-Regret Learning, Portfolio Optimization, and Risk
  – no-regret learning and finance
  – theoretical guarantees and empirical performance
  – incorporating risk: Sharpe ratio, mean-variance, market benchmarks
  – no-regret and option pricing
Outline

• I. Market Microstructure and Optimized Execution
  – online algorithms and competitive analysis
  – reinforcement learning for optimized execution
  – microstructure and market-making

• II. Mechanism Innovation: a Case Study
  – difficult trades and dark pools
  – the order dispersion problem
  – censoring, exploration, and exploitation

• III. No-Regret Learning, Portfolio Optimization, and Risk
  – no-regret learning and finance
  – theoretical guarantees and empirical performance
  – incorporating risk: Sharpe ratio, mean-variance, market benchmarks
  – no-regret and option pricing
Questions of Enduring Interest

• How do (stock) prices “evolve”? How can we model this evolution?
  – classical random walk, diffusion models + drift
    • many recent empirical challenges [Lo & MacKinlay; Brock et al.]
  – autoregressive time series models
    • AR1, ARCH, GARCH, etc. → generalized Ito model
  – computer science: adversarial/worst-case price sequences
    • algorithms analyzed w.r.t. competitive ratios, regret

• Can we design “adaptive” or “learning” algorithms for:
  – executing difficult/large trades?
  – predicting and profiting from movements of prices?

• Models generally ignore market mechanism and liquidity issues
  – at least in part because the data was unavailable and unreliable

• This is changing rapidly… and presents challenges & opportunities
Background on Market Microstructure

- Consider a typical exchange for some specific security
- **Limit order**: specify price (away from the market)
- (Partially) Executable orders are filled immediately
  - prices determined by standing orders in the book
  - one order may execute at multiple prices
- Non-executable orders are placed in the buy or sell **book**
  - sorted by price; top prices are the **bid** and **ask**
- **Market order**: limit order with an extreme price
- Full order books visible in real time
- What are they good for?
Optimized Trade Execution

• Canonical execution problem: sell $V$ shares in $T$ time steps
  – must place market order for any unexecuted shares at time $T$
  – also known as “one-way trading” (OWT)
  – trade-off between price, time… and liquidity

• Problem is ubiquitous
• Multiple performance criteria:
  – Maximum Price:
    • compare revenue to max execution price in hindsight
    • $O(\log(R))$ competitive ratios in infinite liquidity, adversarial price model
      – $R$ = a priori bound on ratio of max to min execution price
      – [El-Yaniv, Fiat, Karp & Turpin]
  – Volume Weighted Average Price (VWAP):
    • compare to per-share average price of executions in hindsight
    • widely used on Wall Street; reduces risk sources to execution
    • by definition, must track prices and volumes
  – Implementation Shortfall:
    • compare per-share price to mid-spread price at start of trading interval
    • an unrealizable ideal
An Online Microstructure Model

- **Market** places a sequence of price-volume limit orders:
  - \( M = (p_1, v_1), (p_2, v_2), \ldots, (p_T, v_T) \) (+ order types)
  - possibly adversarial; also consider various restrictions
  - need to assume bound on \( p_{\text{max}}/p_{\text{min}} = R \)

- **Algorithm** is allowed to interleave its own limit orders:
  - \( A = (q_1, w_1), (q_2, w_2), \ldots, (q_T, w_T) \) (+ order types)

- **Merged sequence** determines executions and order books:
  - \( \text{merge}(M, A) = (p_1, v_1), (q_1, w_1), \ldots, (p_T, v_T), (q_T, w_T) \)
  - assuming zero latency
  - now have complex, high-dimensional state
    - how to simplify/summarize?
What Can Be Done?

[Kakade, K., Mansour, Ortiz ACM EC 2004]

• **Maximum Price:**
  - $O(\log(R))$ infinite liquidity model $\Rightarrow O(\log(R)\log(V))$ in limit order model
  - quantifies worst-case market impact of large trades
  - if $p_1 > p_2 > \ldots$ are execution prices, randomly “guess” $\max\{kp_k\}$
  - note: optimal offline algorithm unknown!

• **VWAP:**
  - $O(\log(Q))$ in limit order model
    - $Q = \text{ratio of max to min total executed volume on allowed sequences}$
    - $Q$ often small empirically; can exploit (entropic) distributional features
  - **Better:** trade $V$ over $\geq \gamma V$ executed shares, $\gamma$ is max order size
    - VWAP “with volume” instead of “with time”
  - Can approach competitive ratio of 1 for large $V$!
  - Sketch of algorithm/analysis:
    - divide time into equal (executed) volume intervals $I_1, I_2, \ldots$
    - place sell order for 1 share at $\sim (1-\varepsilon)^k$ nearest VWAP$_j$
    - if all orders executed, are within $(1-\varepsilon)$ of overall VWAP
    - can’t “strand” more than one order at any given price level
    - optimize $\varepsilon$
  - None of these algorithms “look” in the order books!
Limitations of the Book?

• Even offline revenue maximization is NP-complete
  – advance knowledge of sequence of arriving limit orders
  – [Chang and Johnson, WINE 2008]

• Instability of limit order dynamics
  – relative price formation model (market-making, HFT)
  – small “tweaks” to order sequence can cause large changes in macroscopic quantities
  – e.g. VWAP, volume traded
  – “butterfly effects” and discrete chaos
  – [Even-Dar, Kakade, K., Mansour ACM EC 2006]

• What about empirically?
Reinforcement Learning for Optimized Execution

- Basic idea: execution as state-based stochastic optimal control
  - state: time and shares remaining... what else?
  - actions: position(s) of orders within the book
  - rewards: prices received for executions
  - stochastic: because same state may evolve differently in time

- Large-scale application of RL to microstructure

- Related work:
  - Bertsimas and Lo
  - Coggins, Blazejewski, Aitken
  - Moallemi, Van Roy
Experimental Details

[Nevmyvaka, Feng, K. ICML 2006]

- Stocks: AMZN, NVDA, QCOM (varying liquidities)
- Full OB reconstruction from historical data
- $V = 5K$ and $10K$ shares
  - divided into 1, 4 or 8 levels of observed discretization
- $T = 2$ and $8$ mins
  - divided into 4 or 8 decision points
- Explored a variety of OB state features
- Learned optimal strategy on 1 year of INET training data
- Tested strategy on subsequent 6 months of test data
- Objective function:
  - basis points compared to all shares traded at initial spread midpoint
    - implementation shortfall; an unattainable ideal (infinite liquidity assumption)
- Same basic RL framework can be applied much more broadly
  - e.g. “market-making” strategies [Chan, Kim, Shelton, Poggio]
A Baseline Strategy: Optimized Submit-and-Leave

- Shortfall vs. Limit Price
- Risk vs. Limit Price
- Efficient Frontier

Deep in OB, M.O. at start

[Nevmyvaka, K., Papandreou, Sycara IEEE CEC 2005]
Private State Variables Only: Time and Inventory Remaining

Average Improvement Over Optimized Submit-and-Leave

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>T=4 I=1</td>
<td>27.16%</td>
<td>T=8 I=1</td>
<td>31.15%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T=4 I=4</td>
<td>30.99%</td>
<td>T=8 I=4</td>
<td>34.90%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T=4 I=8</td>
<td>31.59%</td>
<td>T=8 I=8</td>
<td>35.50%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Improvement From Order Book Features

<table>
<thead>
<tr>
<th>Metric</th>
<th>Improvement</th>
<th>Metric</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bid Volume</td>
<td>-0.06%</td>
<td>Ask Volume</td>
<td>-0.28%</td>
</tr>
<tr>
<td>Bid-Ask Volume Misbalance</td>
<td>0.13%</td>
<td>Bid-Ask Spread</td>
<td>7.97%</td>
</tr>
<tr>
<td>Price Level</td>
<td>0.26%</td>
<td>Immediate Market Order Cost</td>
<td>4.26%</td>
</tr>
<tr>
<td>Signed Transaction Volume</td>
<td>2.81%</td>
<td>Price Volatility</td>
<td>-0.55%</td>
</tr>
<tr>
<td>Spread Volatility</td>
<td>1.89%</td>
<td>Signed Incoming Volume</td>
<td>0.59%</td>
</tr>
<tr>
<td>Spread + Immediate Cost</td>
<td>8.69%</td>
<td>Spread+ImmCost+Signed Vol</td>
<td>12.85%</td>
</tr>
</tbody>
</table>
Microstructure and Market-Making

• Canonical market-making:
  – always maintain outstanding buy & sell limit orders; can adjust spread
  – if a buy-sell pair executed, earn the spread
  – only one side executed: accumulation of risk/inventory
  – may have to liquidate inventory at a loss at market close

• A simple model, algorithm and result:
  – price time series \( p_0, \ldots, p_T \), where \( d_t = |p_{t+1} - p_t| < D \), infinite liquidity
  – algorithm maintains ladder of matched order pairs up to depth \( D \)
  – let \( z = p_T - p_0 \) (global price change) and \( K = \sum_t d_t \) (sum of local changes)
  – then profit = \( K - z^2 \)
  – +/-1 random walk (Brownian): profit = 0
  – but profit > 0 on any “mean-reverting” time series
  – [Chakraborty and K., ACM EC 2011]

• Learning and market-making: Sanmay Das and colleagues
Outline

• I. Market Microstructure and Optimized Execution
  – online algorithms and competitive analysis
  – reinforcement learning for optimized execution
  – microstructure and market-making

• II. Mechanism Innovation: a Case Study
  – difficult trades and dark pools
  – the order dispersion problem
  – censoring, exploration, and exploitation

• III. No-Regret Learning, Portfolio Optimization, and Risk
  – no-regret learning and finance
  – theoretical guarantees and empirical performance
  – incorporating risk: Sharpe ratio, mean-variance, market benchmarks
  – no-regret and option pricing
Modern “Light” Exchanges

**Major disadvantage:** executing very large orders
* distributing over time and venues insufficient
* many buy-side parties are “compelled”
Thus the advent of... **Dark Pools**
* specify side and volume only
* no price, execution by time priority
* price generally pegged to light midpoint
* not seeking price *improvement*, just execution
* only learn (partial) fill for your order
SEC Weighs New Regulations for Dark Pools

By SARAH N. LYNCH

WASHINGTON — The Securities and Exchange Commission unanimously agreed Wednesday to consider three proposals aimed at shedding more light on non-public electronic trading entities including dark pools, which match big stock orders privately.

The proposals would require dark pools to make information about an investor's interest in buying or selling a stock available to the public instead of only sharing it with a select group operating with a dark pool. They would also require dark pools to publicly identify if their pool executed a trade.

"We should never underestimate or take for granted the wide spectrum of benefits that come from transparency," SEC Chairman Mary Schapiro said. "Transparency plays a vital role in promoting public confidence in the honesty and integrity of financial markets."

Dark pools, a type of alternative trading system that doesn’t display quotes to the public, are just one part of a broader probe the SEC is conducting into market structures. Recently, the SEC also voted to consider banning flash orders, which let some traders get a sneak peek at market activity. The agency is also looking into other areas.
TORA Crosspoint
Instinet
SmartPool
Posit/MatchNow from Investment Technology Group (ITG)
Liquidnet
NYFIX Millennium
Pulse Trading BlockCross
RiverCross
Pipeline Trading Systems
Barclays Capital - LX Liquidity Cross
BNP Paribas
BNY ConvergEx Group
Citi - Citi Match
Credit Suisse - CrossFinder
Fidelity Capital Markets
GETCO - GETMatched
Goldman Sachs SIGMA X
Knight Capital Group - Knight Link, Knight Match
Deutsche Bank Global Markets - DBA(Europe), SuperX ATS (US)
Merrill Lynch – MLXN
Morgan Stanley
Nomura - Nomura NX

UBS Investment Bank
Ballista ATS Ballista Securities LLC
BlocSec[citation needed]
Bloomberg Tradebook (an affiliate of Bloomberg L.P.)
Daiwa – DRECT
BIDS Trading - BIDS ATS
LeveL ATS
International Securities Exchange
NYSE Euronext
BATS Trading
Direct Edge
Swiss Block
Nordic@Mid
Chi-X
Turquoise
Bloomberg Tradebook
Fidessa - Spotlight
SuperX+ – Deutsche Bank
ASOR – Quod Financial
Progress Apama
ONEPIPE – Weeden & Co. & Pragma Financial
Xasax Corporation
Crossfire – Credit Agricole Cheuvreux
The Dark Pool (Allocation) Problem

Given a sequence or distribution of “client” or parent orders, how should we distribute the desired volumes over a large number of dark pools? (a.k.a. Smart Order Routing (SOR))

May initially know little about relative quality/properties of pools
* may be specific to name, volatility, volume,…
* ...a learning problem
* (related to “newsvendor problem” from OR)

To simplify things, will generally assume:
* client orders all on one side (e.g. selling)
* client orders come i.i.d. from a fixed distribution
  ...even though our “child” submissions to pools will not be i.i.d.
* statistical properties of a given pool are static

All can be relaxed in various ways, at the cost of complexity
Modeling Available Volume: Single Pool

\[ P[s] \]

v shares submitted

\[ \text{draw } s \sim P \]

execute \[ \text{min}(v, s) \]

censored observations
Multiple Pools

Client volume $V$

Allocate... 
...How?
A Statistical Sub-Problem

From a given pool \( P[s] \), we observe a sequence of censored executions. At time \( t \), we submitted \( v(t) \) shares and \( s(t) \leq v(t) \) were executed.

Q: What is the maximum likelihood estimate of \( P[s] \)?
A: The Kaplan-Meier estimator from biostatistics and survival analysis:
* start with empirical distribution of uncensored observations
* process censored observations from largest to smallest
* distribute over larger values proportional to their current weight

Known to converge to \( P[s] \) asymptotically under i.i.d. submissions
* also need support conditions on submission distribution
* for us, i.i.d. violated by dependence between venue submissions

Can prove and use a stronger lemma (paraphrased):
* for any volume \( s \), \( |P[s] - P'[s]| \sim 1/\sqrt{N(s)} \)
* \( N(s) \sim \) number of times we have submitted \( > s \) shares

For analysis only, define a cut-off \( c[i] \) for each venue distribution \( P_i \):
* we “know” \( P_i[s] \) accurately for \( s \leq c[i] \)
* may know little or nothing above \( c[i] \)
The Learning Algorithm and Analysis

[Ganchev, K., Nevmyvaka, Wortman UAI, CACM]

Algorithm:
- initialize estimated distributions $P'_1, P'_2, \ldots, P'_k$
- repeat:
  * compute greedy optimal allocations to each venue given the $P'_i$
  * use censored executions to re-estimate $P'_i$ using optimistic K-M

Analysis:
- if allocation to every venue $i$ is $< c[i]$, already near-optimal;
  know “enough” about the $P_i$ to make this allocation (“exploit”)
- if for some venue $j$, submitted volume $> c[j]$, we “explore”;
  so eventually $c[j]$ will increase $\rightarrow$ improve $P'_j$
- optimistic: slight tail modification ensures always exploit/explore
- analogy to $E^3$/RMAX family for RL

Main Theorem: algorithm efficiently converges to near-optimal
  * non-parametric and parametric versions
Algorithm vs. Uniform Allocation

Fraction Executed

Order Half-life

learning vs. uniform allocation

small orders
large orders
Algorithm vs. Ideal Allocation

Fraction Executed

Order Half-life

learning vs. ideal allocation for small and large orders.
Algorithm vs. Bandits

* Nice no-regret follow-up: Agarwal, Barlett, Dama AISTATS 2010
Outline

• I. Market Microstructure and Optimized Execution
  – online algorithms and competitive analysis
  – reinforcement learning for optimized execution
  – microstructure and market-making

• II. Mechanism Innovation: a Case Study
  – difficult trades and dark pools
  – the order dispersion problem
  – censoring, exploration, and exploitation

• III. No-Regret Learning, Portfolio Optimization, and Risk
  – no-regret learning and finance
  – theoretical guarantees and empirical performance
  – incorporating risk: Sharpe ratio, mean-variance, market benchmarks
  – no-regret and option pricing
Basic Framework

• An underlying universe of K assets $U = \{S_1, \ldots, S_K\}$
• Goal: manage a “profitable” portfolio over $U$
  – each trading period $S_i$ grows/shrinks $q_i = (1+r_i), r_i \in [-1,\infty]$  
  – we maintain a distribution $w$ of wealth, fraction $w_i$ in $S_i$
  – all quantities indexed (superscripted) by time $t$
• Traditionally: K assets are long positions in common stocks
• More generally: K assets are any collection of investment instruments:
  – long and short positions in common stocks, cash, futures, derivatives
  – technical trading strategies, pairs strategies, etc.
  – generally need instruments/performance to be “stateless”: can be entered at any time
• How do we measure performance relative to $U$?
  – average return (~“the market”): place $1/K$ of initial wealth in each $S_i$ and leave it there
  – Uniform Constant Rebalanced Portfolio (UCRP): set $w_i = 1/K$ and rebalance every period
    • exponential growth (factor $9/8$) on $S_1 = (1,1,1,1,\ldots)$ and $S_2 = (2,1/2,2,1/2,\ldots)$; reversion effect
  – Best Single Stock (BSS) in hindsight
  – Best Constant Rebalanced Portfolio (BCRP) in hindsight
  – Note: must place some restrictions on comparison class
Online Algorithms: Theory

- Assume nothing about sequence of returns $r_i$ (except maybe max loss)
- On arbitrary sequence $r^1, \ldots r^T$, algorithm $A$ dynamically adjusts portfolio $w^1, \ldots, w^t$
- Compare cumulative return of $A$ to BSS or BCRP (in hindsight)
- Powerful families of no-regret algorithms: for all sequences,
  - $\frac{\text{Return}(A)}{T} \geq \frac{\text{Return}(\text{BSS})}{T} - O(\sqrt{\log(K)/T})$
  - or $\frac{\log(A's \text{ wealth})}{T} \geq \frac{\log(\text{BCRP wealth})}{T} - O(K/T)$ (Cover’s algorithm; exponential growth)
  - “complexity penalty” for large $K$; per-step regret is vanishing with $T$
- How is this possible?
  - note: for this to be interesting, need BSS or BCRP to strongly outperform the average
Cover’s Algorithm

• K stocks, T periods
• \( W_t(p) = \) wealth of portfolio/distribution \( p \) after \( t \) periods
• Invest initial wealth uniformly across all CRPs and leave it
• Equivalent:
  – initial portfolio \( p_1 = (1/K, \ldots, 1/K) \)
  – \( p_{t+1} = \frac{\int_{p} W_t(p)p \, dp}{\int_{p} W_t(p) \, dp} \)
• Learning at the stock level, but not at the portfolio level!
• Now let \( p^* \) maximize \( W(p^*) = W_T(p^*) \) (BCRP in hindsight)
• Then for any \( c: W(A) \geq r^K (1-r)^T W(p^*) \)
  – \( r^K: \) amount of weight in \( r \)-ball around \( p^* \)
  – \( (1-r)^T: \) if \( p \) is within \( r \) of \( p^* \), must make at worst factor \( (1-r) \) less at each period
• Picking \( r = 1/T: W(A) \geq (1/T)^K (1 - 1/T)^T W(p^*) \sim (1/T)^K W(p^*) \)
• So \( \log W(A) \geq \log W(p^*) - K \log T \)
• Only interesting for exponential growth
Tractable Algorithms

• Most update weights multiplicatively, not additively
• Flavor of a typical algorithm (e.g. Exponential Weights):
  – \( w_i \leftarrow \exp(\eta r_i)w_i \), renormalize
• One (crucial) parameter: learning rate \( \eta \)
  – for the theory, need to optimize \( \eta \sim 1/\sqrt{T} \)
  – generally are assuming momentum rather than mean reversion
  – note: \( \eta = 0 \) (no learning) is UCRP; a form of mean reversion
  – value of \( \eta \) also strongly influences portfolio concentration \( \rightarrow \) variance/risk
• Let’s look at some empirical performance
Data Period: early 2005 – end 2011 (~7 years)
Underlying Instruments: stocks in S&P 500 (selection bias)
Daily (closing) returns
Wealth of investing $1 in each stock
long positions only
UCRP: magenta
Cover’s algorithm: red
Exponential Weights (optimized): green
long and short position
UCRP long only: magenta
UCRP short only: yellow
Cover’s algorithm: red
Exponential Weights: green
What About Risk?

- Sharpe Ratio = (mean of returns)/(standard deviation of returns)
- Mean-Variance (MV) criterion = mean – variance
- Maximum Drawdown; Value at Risk (VaR)
- Concentration limits
- Market index/average as a lower bound
Some Relevant Theory

- What about no regret compared to BSS in hindsight w.r.t. risk-return metrics?
  - e.g. BSS Sharpe, BSS M-V,…
  - can prove any online algorithm must have constant regret…
  - …in fact, even offline competitive ratio must be constant
  - variance constraints introduce switching costs or state
  - [Even-Dar, K., Wortman ALT 2006]

- But can preserve traditional no-regret with benchmarking to average
  - additive reward setting
  - guarantee O(\sqrt{T}) cumulative regret to best, O(1) to average
  - Idea: only increase \( \eta \) as data “proves” best will beat average
  - worst case: track the market
  - [Even-Dar, K, Mansour, Wortman COLT 2007]

- “State” generally ruins no-regret theory
- Lots of room for innovation/improvement
No-Regret and Option Pricing

- Option (European call): right, but not obligation, to purchase shares at a fixed price and future time
- E.g. AAPL now trading ~$546; option to purchase at $600 in a year
- Option should cost *something* --- but what?
  - depends on uncertainty/fluctuations
- Black-Scholes:
  - assume future price evolution follows geometric Brownian motion
  - B: borrow money to buy options now; if options “in the money”, exercise and pay back loan
  - S: sell options now for cash; if options in the money, pay counterparty
  - correct option price: neither B nor S has positive expected profit
- What if the future price evolution is *arbitrary*?
- DeMarzo, Kremer, Mansour STOC 06:
  - hedging strategy that has no regret to option payoff
  - multiplicative weight update algorithm
- Abernethy, Frongillo, Wibisono STOC 2012:
  - view option pricing as an adversarial game
  - minimax price is same as Black-Sholes under Brownian motion!
- More complex derivatives with asymmetric info may be intractable to price
  - “pay 1$ if AAPL price increases x% where x matches last two digits of a prime factor of N”
  - intractability of planted dense subgraph $\rightarrow$ difficulty in pricing natural derivatives (e.g. CDS)
  - Arora, Barak, Brunnermeier, Ge
Conclusions

• Many algorithmic challenges in modern finance
• Lower level: market microstructure, optimized execution metrics & problems
• Higher level: portfolio optimization, option pricing, no-regret algorithms
• New market mechanisms lead to new algorithmic challenges (e.g. dark pools)