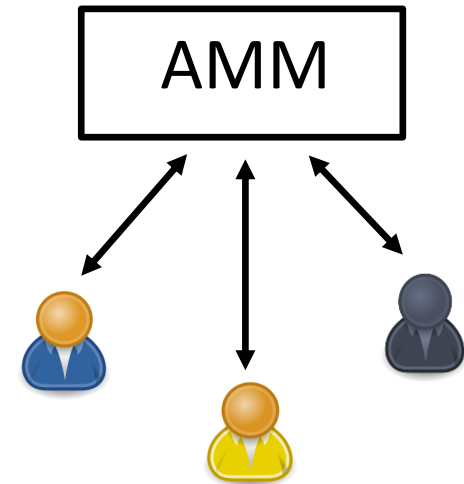

Designing Automated Market Makers for Combinatorial Securities: A Geometric Viewpoint

Prommy Sultana Hossain , Xintong Wang, Fang-Yi Yu

Outline

- Preliminary of prediction market
 - Automatic market maker and LSMR
 - Large outcome space: combinatorial securities
- AMM as a range query problem
- AMM for de-fi

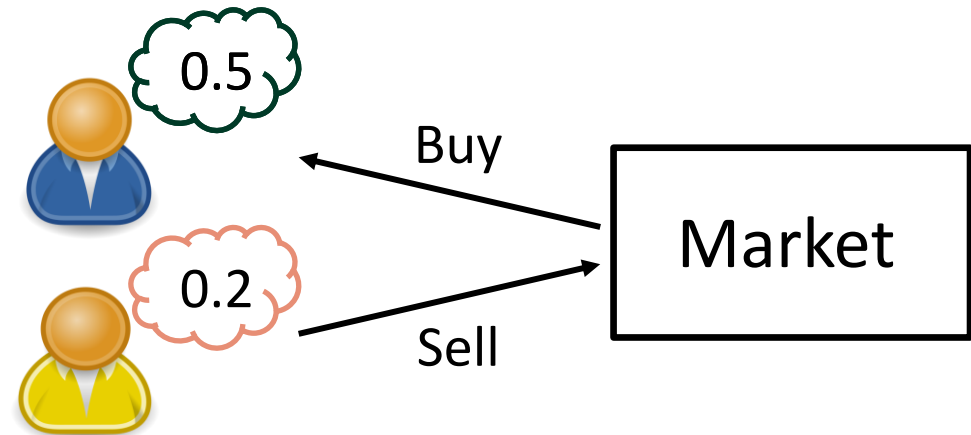


Why Can Markets Aggregate Information?

- Offer securities whose payoff is tied to outcomes of an event.

\$1 if rain on Jan 15, \$0 otherwise

\$0.3



- A market maker adjust the price to “equilibrium”
- $\text{Price} \approx \text{Pr}[\text{event} \mid \text{all information}]$

Examples of Prediction Markets



In 2020

Q1, 2021

Maybe later



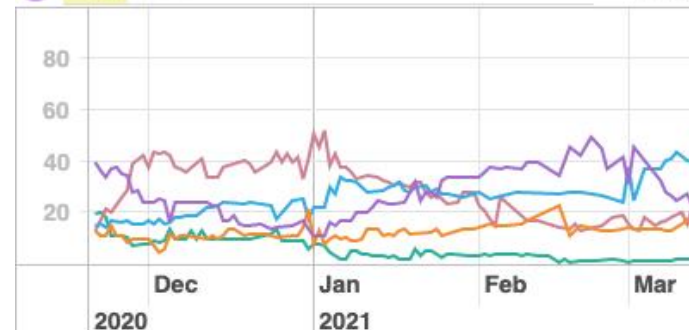
Q1, 2021 (or before)

Q2, 2021

Q3, 2021

Q4, 2021

Maybe later



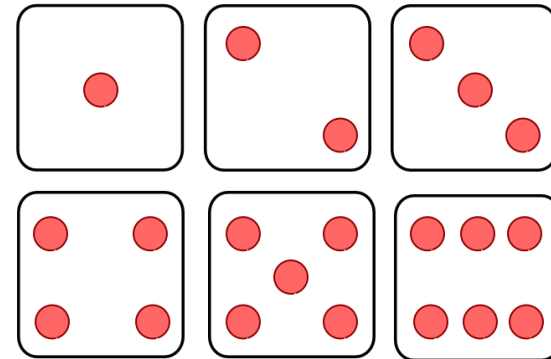
How many tweets will @realDonaldTrump post from noon Nov. 13 to 20?

Contract	Latest Yes Price	Best Offer	Buy Yes	Buy No	Best Offer
179 or fewer	2¢ NC	3¢	Buy Yes	Buy No	98¢
180 - 189	2¢ NC	3¢	Buy Yes	Buy No	98¢
190 - 199	3¢ 1¢↓	4¢	Buy Yes	Buy No	97¢
200 - 209	9¢ 1¢↑	10¢	Buy Yes	Buy No	92¢
210 - 219	12¢ 1¢↑	12¢	Buy Yes	Buy No	90¢
220 - 229	13¢ 2¢↑	13¢	Buy Yes	Buy No	89¢
230 - 239	15¢ 1¢↑	15¢	Buy Yes	Buy No	86¢
240 - 249	18¢ 4¢↑	18¢	Buy Yes	Buy No	83¢
250 or more	35¢ 10¢↓	39¢	Buy Yes	Buy No	64¢

AMM for prediction market

- Shares $w = (w_1, \dots, w_n)$
- Each security $i = 1, \dots, n$ has
 - An outcome and pay 1\$ if happens
 - Price $p_i(w)$
- Experts: buy s shares of security i and pay
- Market maker (AMM): change prices based on shares

Security	1	2	3	4	5	6
Share w_i	1	1	1	1	2	2
Price p_i	1/8	1/8	1/8	1/8	1/4	1/4



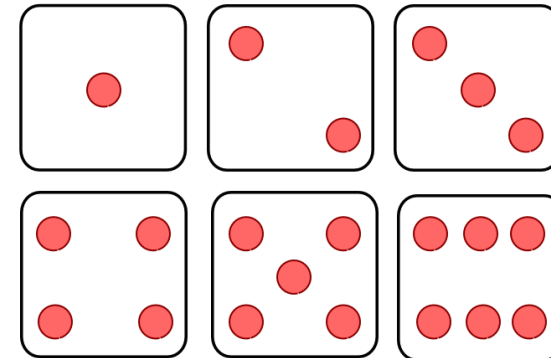
$n = 6$

Logarithmic Market Scoring Rule

Let $C(w) = \log_2 \sum_{i=1}^n 2^{w_i}$ and $w = (w_1, \dots, w_n)$

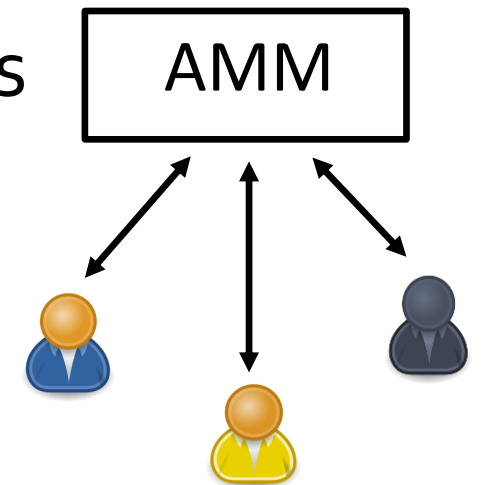
- Each security i has
 - An outcome and pay 1\$ if happens
 - Price $p_i(w) = \frac{2^{w_i}}{\sum_j 2^{w_j}} = \frac{\partial C(w)}{\partial w_i}$
- Experts: buy s shares of security i and pay $C(w + s\delta_i) - C(w)$

Security	1	2	3	4	5	6
Share w_i	1	1	1	1	2	2
Weight 2^{w_i}	2	2	2	2	4	4
Price p_i	1/8	1/8	1/8	1/8	1/4	1/4



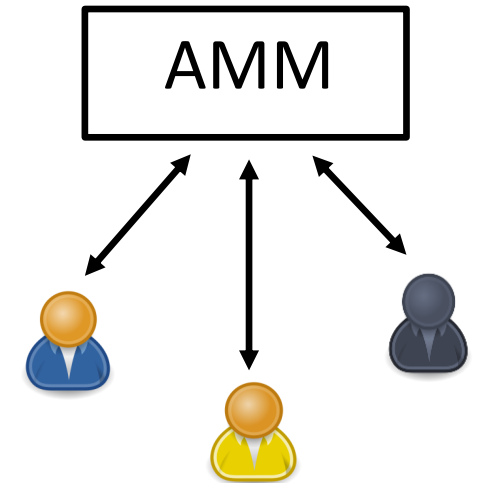
LMSR as an online algorithm problem

- Given a set of outcomes and initial shares $w^{(0)}$, in each round t
 1. Query price(i): return $p_i(w^{(t)})$
 2. Query cost(i, s): return $C(w^{(t)} + s\delta_i) - C(w^{(t)})$
 3. Query buy(i, s): update $w^{(t+1)} = w^{(t)} + s\delta_i$
- Design a data structure to support above queries



Outline

- Preliminary of prediction market
 - Automatic market maker and LSMR
 - Large outcome space: combinatorial securities
- AMM as a range query problem
- AMM for de-fi



Prediction market

- Given a random variable on \mathcal{X} , trade securities with an expert
 - Each security has
 - An outcome and pay 1\$ if happens
 - Price
 - Experts: buy securities
 - AMM: change prices based on shares

2024 Election Forecast

Live and accurate forecasts by the world's largest prediction market.

Election in 27 DAYS 11 HRS 17 MIN 28 SEC

Presidency Senate

How it works Share Embed



Trump

53.2%

▲ 0.6%



Polymarket
polymarket.com/elections

46.2%

▼ 0.5%



Harris

$\mathcal{X} = \{\text{Harris wins, Trump wins}\}$

Large outcome space

- Given a random variable on \mathcal{X} , trade securities with an expert
 - Each security has
 - An outcome and pay 1\$ if happens
 - Price
 - Experts: buy securities
 - AMM: change prices based on shares

2024 Election Forecast

Live and accurate forecasts by the world's largest prediction market.

Election in 27 DAYS 11 HRS 17 MIN 28 SEC

Presidency Senate

How it works Share <> Embed



Trump

53.2%

▲ 0.6%



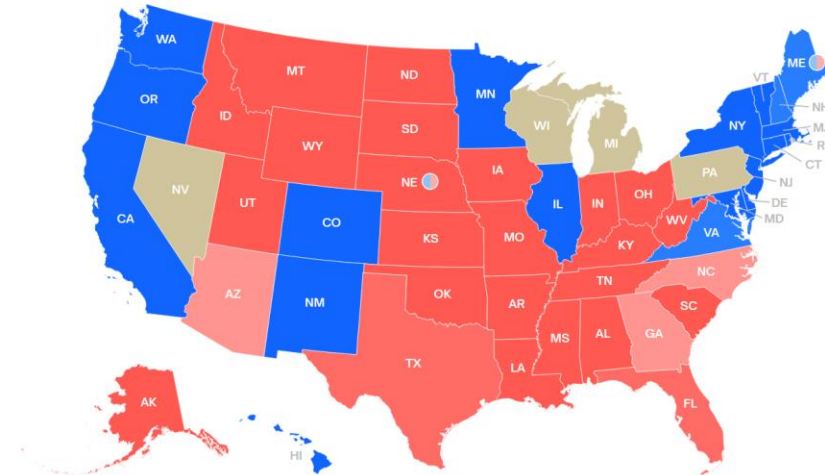
Polymarket
polymarket.com/elections

46.2%

▼ 0.5%



Harris



$\mathcal{X} = \{\text{Each state's winner}\}$

Large outcome space

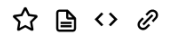
$$\mathcal{X} = \mathbb{R}$$

- Given a random variable on \mathcal{X} , trade securities with an expert
 - Each security has
 - An outcome and pay 1\$ if happens
 - Price
 - Experts: buy securities
 - AMM: change prices based on shares



What price will Bitcoin hit in 2025?

\$591,750 Vol. ⌚ Dec 31, 2025



OUTCOME	% CHANCE ↺		
\$1,000,000 \$103,025 Vol. 📦	4%	Buy Yes 4.6¢	Buy No 95.8¢
\$250,000 \$25,090 Vol. 📦	14%	Buy Yes 15¢	Buy No 87¢
\$200,000 \$53,124 Vol. 📦	23%	Buy Yes 24¢	Buy No 78¢
\$150,000 \$37,053 Vol. 📦	43%	Buy Yes 44¢	Buy No 59¢
\$130,000 \$13,674 Vol. 📦	66%	Buy Yes 69¢	Buy No 38¢
\$120,000 \$11,516 Vol. 📦	74%	Buy Yes 77¢	Buy No 29¢
\$110,000 \$15,618 Vol. 📦	85%	Buy Yes 86¢	Buy No 17¢

Combinatorial prediction market

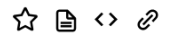
$$\mathcal{X} = \mathbb{R}$$

- Given a random variable on \mathcal{X} , trade securities with an expert
 - Each security in \mathcal{F} has
 - A **subset of outcomes** and pay 1\$ if happens
 - Price
 - Experts: buy securities
 - AMM: change prices based on shares



What price will Bitcoin hit in 2025?

\$591,750 Vol. ⌚ Dec 31, 2025



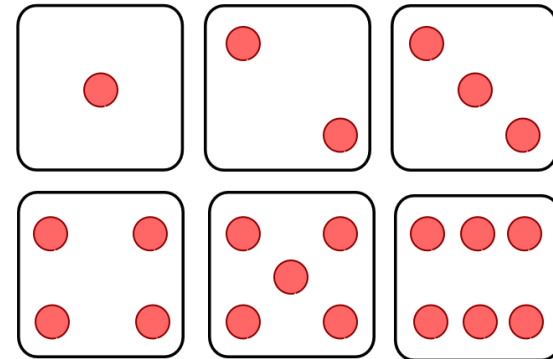
OUTCOME	% CHANCE ↕		
\$1,000,000 \$103,025 Vol. 📦	4%	Buy Yes 4.6¢	Buy No 95.8¢
\$250,000 \$25,090 Vol. 📦	14%	Buy Yes 15¢	Buy No 87¢
\$200,000 \$53,124 Vol. 📦	23%	Buy Yes 24¢	Buy No 78¢
\$150,000 \$37,053 Vol. 📦	43%	Buy Yes 44¢	Buy No 59¢
\$130,000 \$13,674 Vol. 📦	66%	Buy Yes 69¢	Buy No 38¢
\$120,000 \$11,516 Vol. 📦	74%	Buy Yes 77¢	Buy No 29¢
\$110,000 \$15,618 Vol. 📦	85%	Buy Yes 86¢	Buy No 17¢

Examples of combinatorial securities $(\mathcal{X}, \mathcal{F})$

- Continuous
 - Bitcoin price
 - Intervals, e.g., $[110k, +\infty]$, $[120k, +\infty]$
 - Stock market
 - 2D intervals: the opening value of AMZN and GOOGL at 4pm tomorrow
 - Hyperplane: Index funds (S&P500)
- Discrete
 - Presidential election
 - Swing states, Popular votes
 - Sport betting
 - Playoff, championship

Traditional Market Implementation

- Independent markets
 - Logic inconsistent
 - Arbitrage
- Integrated combinatorial prediction market for $(\mathcal{X}, \mathcal{F})$
 - Computational complexity [CFNP07]
 - Algorithms for Intervals [DWPR21]



Our Contributions

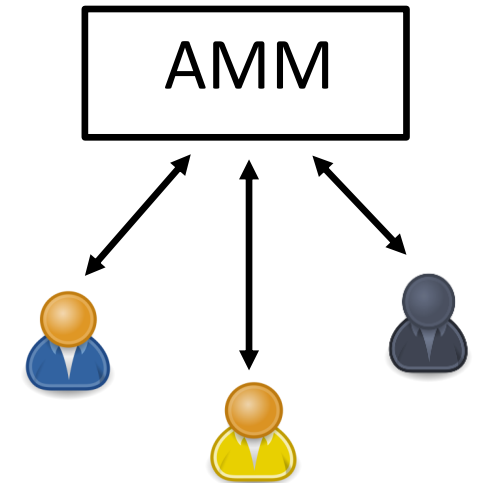
AMM for combinatorial securities = range query problem

- Partition tree algorithms for LMSR
- Generalize to quadratic scoring rule and 3/2-power scoring rule
- Incorporate multi-resolution market designs into a partition-tree scheme.
- Combinatorial Swap in DeFi

Set systems $(\mathcal{X}, \mathcal{F})$	Running time
Intervals (Example 2.3)	$\Theta(\log n)$
d -orthogonal sets (Example 2.4)	$O(n^{1-1/d})$
Hyperplane in \mathbb{R}^d (Example 2.5)	$O(n^{1-1/d})$
Finite VC (Example 2.6)	$O(n^{1-\epsilon})$ with $\epsilon > 0$
Infinite VC (Examples 2.7 and 2.8)	no $o(n)$

Outline

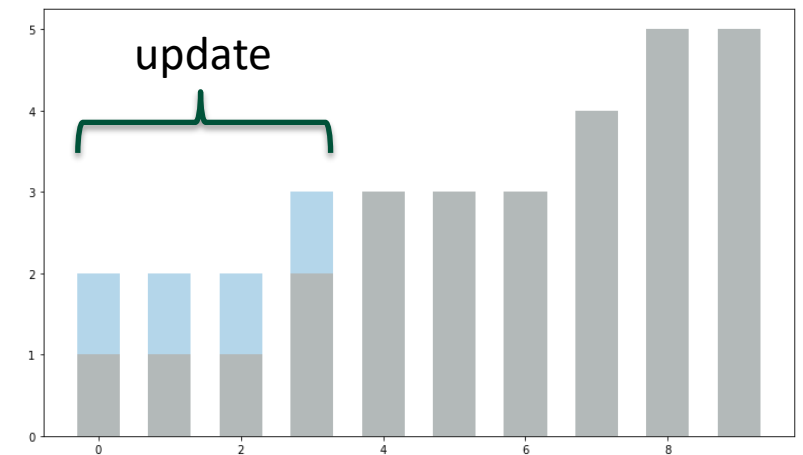
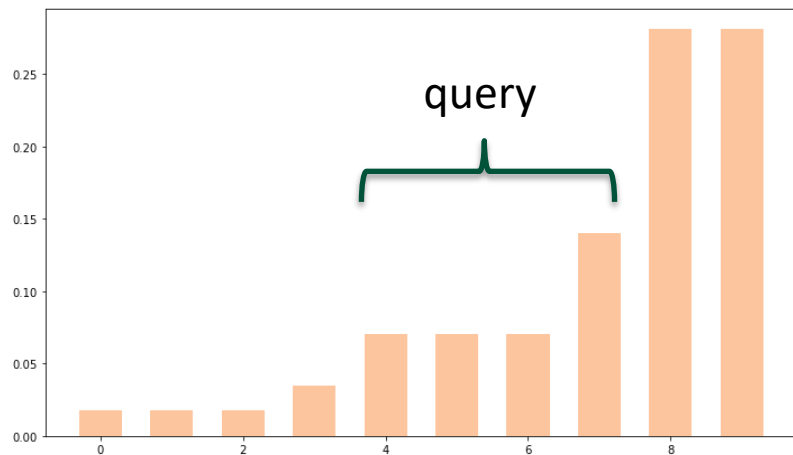
- Preliminary of prediction market
 - Automatic market maker and LSMR
 - Large outcome space: combinatorial securities
- AMM as a range query problem
- AMM for de-fi



Range query range update (RQRU)

Given $(\mathcal{X}, \mathcal{F})$ and initial weights $W^{(0)}: \mathcal{X} \rightarrow \mathbb{R}_+$, RQRU requests a sequence of operations: for any $E \in \mathcal{F}$ and $S \in \mathbb{R}_+$:

- $\text{query}(E; W)$: return the total weight of range E , $\sum_{x \in E} W(x)$.
- $\text{update}(E, S; W)$: update $W(x) \leftarrow \begin{cases} S \cdot W(x) & \text{if } x \in E \\ W(x) & \text{otherwise} \end{cases}$



Idea

Given \mathcal{F} , $C(w) = \log_2 \sum_i 2^{w_i}$ and w

- Each security has
 - An event E in \mathcal{F} and pay 1\$ if happens
 - Price $p_E(w) = \frac{\sum_{i \in E} 2^{w_i}}{\sum_j 2^{w_j}}$
- Experts: buy s shares of security E and pay $C(w + s1_E) - C(w)$

Idea: price=query

Given \mathcal{F} , $C(w) = \log_2 \sum_i 2^{w_i}$ and w

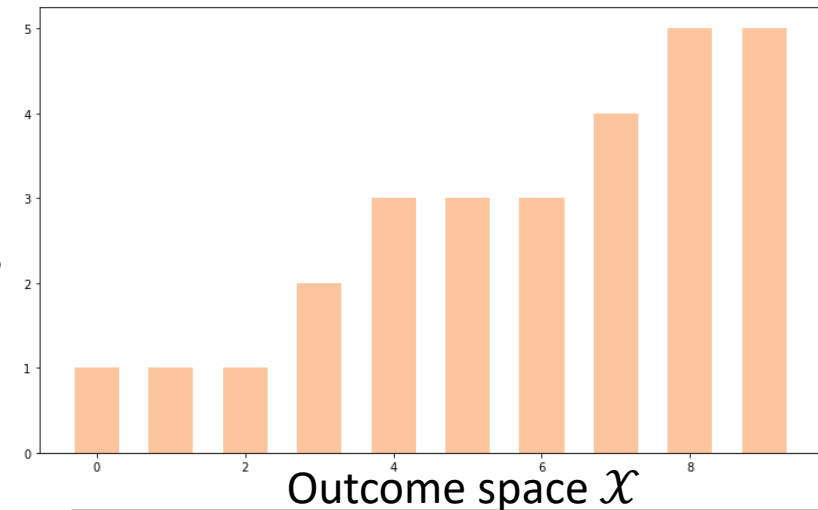
- Each security has
 - An event E in \mathcal{F} and pay 1\$ if happens

query

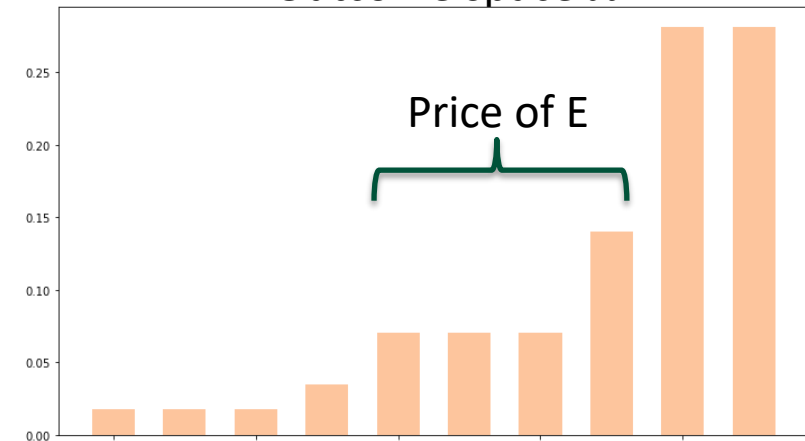
- Price $p_E(w) = \sum_{i \in E} \frac{2^{w_i}}{\sum_j 2^{w_j}}$

- Experts: buy s shares of security E and pay $C(w + s1_E) - C(w)$

Number
of shares



Price

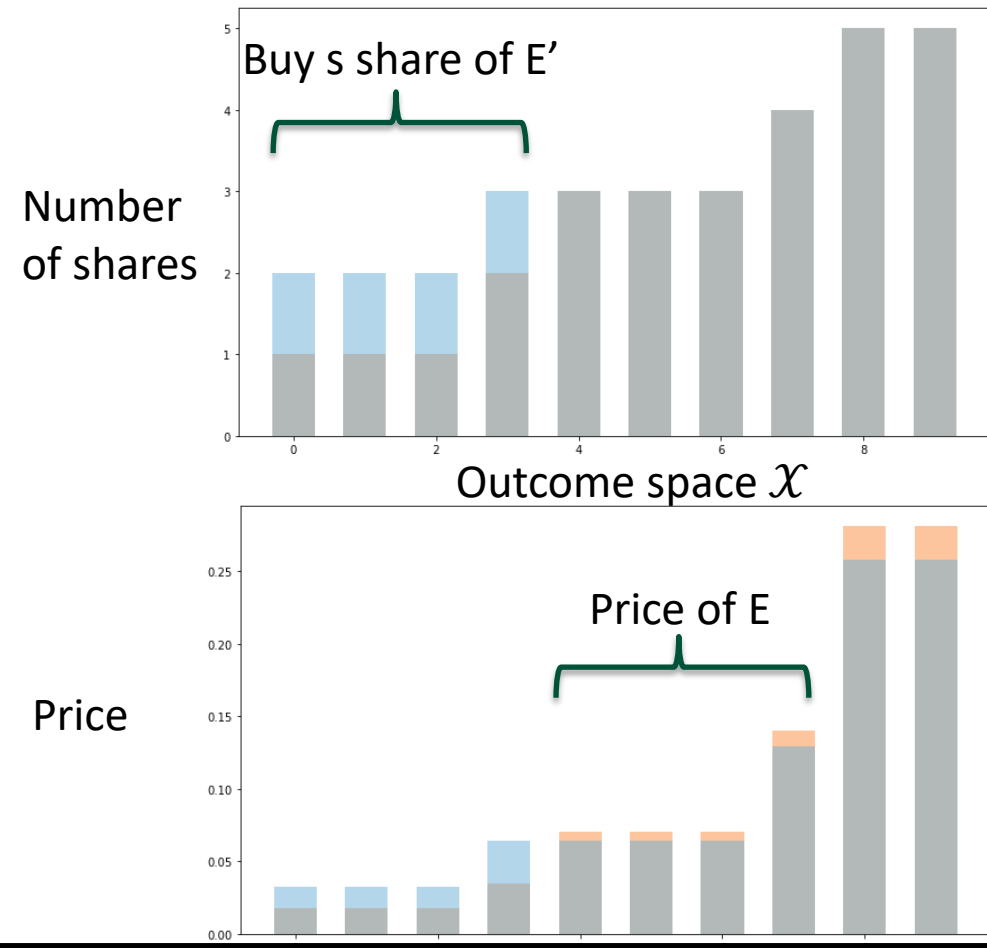


Idea: buy=update

Given \mathcal{F} , $C(w) = \log_2 \sum_i 2^{w_i}$ and w

- Each security has
 - An event E in \mathcal{F} and pay 1\$ if happens
 - Price $p_E(w) = \sum_{i \in E} \frac{2^{w_i}}{\sum_j 2^{w_j}}$
- Experts: buy s shares of security E and pay $C(w + s1_E) - C(w)$

update



AMM = range query problem

AMM

- Given \mathcal{X} and securities \mathcal{F} , in each round
 - Experts: buy/sale securities $E \in \mathcal{F}$
 - Market maker: update prices

Range query problem

- Given \mathcal{X} and ranges \mathcal{F} , each rounds
 - Update state of the space
 - Query sum of points in a range $E \in \mathcal{F}$

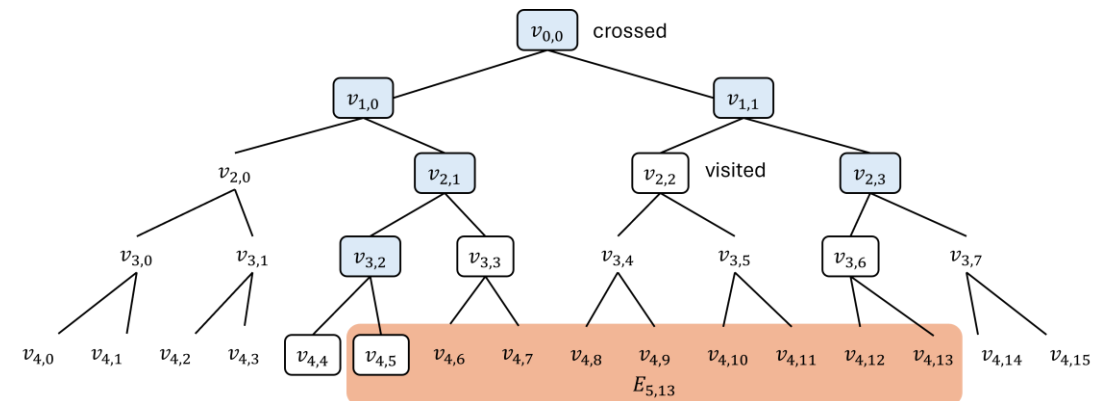
LMSR		RQRU	
Price and cost	$w_i = 2^{w_i}$ \longleftrightarrow	Query (+)	
Buy		Update (*)	

Beyond LMSR

- AMM=RQRU

Scoring rule	Query	Update
Log scoring rule	Addition	Multiplication
Quadratic scoring rule	Addition	Addition
3/2-power scoring rule	Addition	Group action

– multi-resolution market: local arbitrage removal



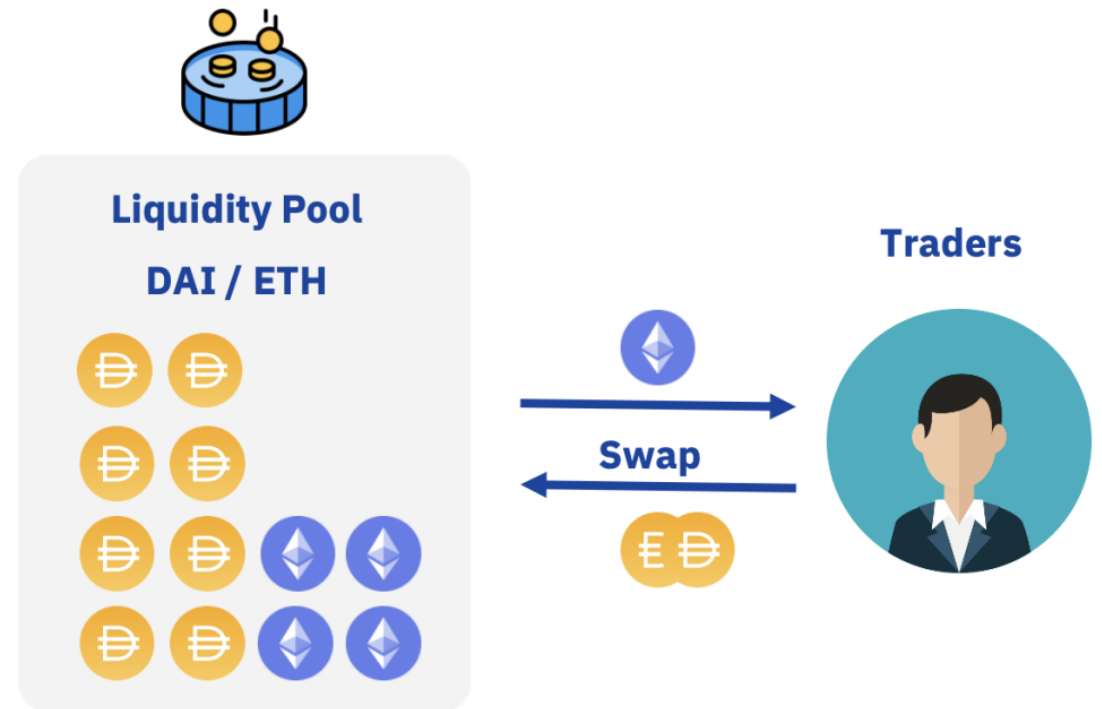
Decentralized exchange market

- Constant function market maker
 - Uniswap, Balancer...
- Combinatorial swap problem = range update query



Constant function market maker

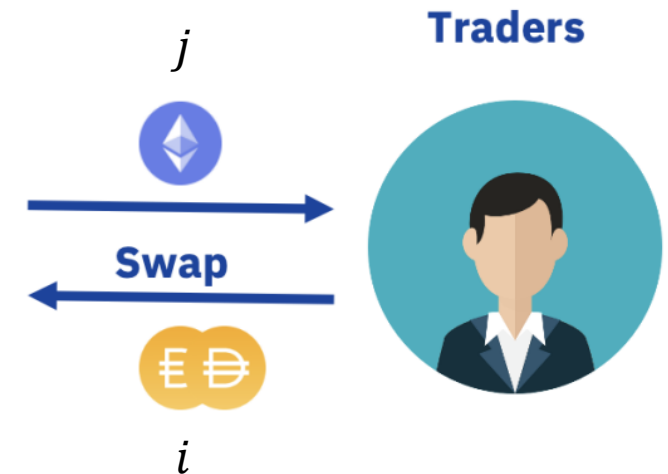
- AMM on states of n assets $w \in \mathbb{R}^n$ with ϕ on \mathbb{R}^n
- A trade $\Delta \in \mathbb{R}^n$ is valid if
$$\phi(w + \Delta) = \phi(w)$$
- E.g. Uniswap (v1/v2) $\phi(w) = \prod_i w_i$ (constant product market maker)



Swap trade

- Given $w \in \mathbb{R}^n$ and constant product market maker ϕ on \mathbb{R}^n , a swap trade $\Delta = s1_i - 1_j \in \mathbb{R}^n$
- A valid s satisfies

$$(w_i + s) = \frac{w_i w_j}{(w_j - 1)}$$



Swap trade for baskets

- Given $w \in \mathbb{R}^n$, $E, E' \subseteq [n]$ and constant product market maker ϕ on \mathbb{R}^n , a swap trade for baskets $\Delta = s1_E - 1_{E'} \in \mathbb{R}^n$

- A valid s satisfies

$$\prod_{i \in E} (w_i + s) = \frac{\prod_{i \in E} w_i \prod_{j \in E'} w_j}{\prod_{j \in E'} (w_j - 1)}$$

- The above can be reduced to RQRU with $*$ query and $+$ update

Summary

- AMM for combinatorial securities = range query problem
 - Sublinear Time LMSR for bounded VC dimension securities
 - Generalize: quadratic scoring rule and $3/2$ -power scoring rule
 - Incorporate multi-resolution market designs into a partition-tree scheme.
 - Combinatorial Swap in DeFi
- Future work
 - Better scoring rules for AMM
 - Approximation and dynamic data structure
 - Multi-resolution
 - CFMM

