An analysis of Uniswap markets
CryptoEconSys 2020, MIT

Guillermo Angeris*, Hsien-Tang Kao, Rei Chiang, Charlie Noyes, Tarun Chitra

March 8, 2020
Outline

The pricing oracle problem

Automated Market Makers

An analysis of Uniswap

Acknowledgements
Measuring the price of assets

- Often, we need a way of measuring the price of an asset
- (Normally) easy: ask how much is someone willing to pay!
- In the traditional setting, this led to order books
Order book methods

- **Bid**: How much an agent is willing to pay for an asset
- **Ask**: How much an agent is willing to sell an asset for
- A trusted party keeps a record of all unfulfilled bids and asks
- When the highest bidder bids more than the lowest asker, the trade is executed
- The price of this trade is the ‘current market price’
Disadvantages

A trusted party keeps a record of all bids and asks
Linear space requirement

When the highest bidder bids more than the lowest asker [...] 
Price may update slowly, esp. with a small number of agents
Outline

The pricing oracle problem

Automated Market Makers

An analysis of Uniswap

Acknowledgements
Automated Market Makers

Savage ’71, Hanson ’02

- **Idea:** use a (simple) formula to determine asset price
- Third-parties pool their assets (say $A$ and $B$) into reserves
- Price set too low: agents purchase reserves at current price
- Price set too high: agents sell to reserves at current price
Automated Market Makers
Savage ’71, Hanson ’02

- **Idea:** use a (simple) formula to determine asset price

- Third-parties pool their assets (say $A$ and $B$) into reserves

- Price set too low: agents purchase reserves at current price

- Price set too high: agents sell to reserves at current price

- Using this idea, set price based on assets remaining in reserves

- *e.g.*, if too much of asset $A$ remains, compared to asset $B$, decrease the price of $A
Automated Market Maker examples

- Simplest example: fixed asset price at all reserve amounts
  *i.e.*, a flat line

- Another example: reported price is ratio of two asset reserves
  This curve is Uniswap!
Uniswap (and constant product markets)

- Constant product markets (e.g., Uniswap) is the family of curves whose reserves $R_{\alpha}, R_{\beta}$ must always satisfy:

$$R_{\alpha} R_{\beta} = k,$$

for some constant $k$ (no fees)

- In this case, we will assume that $\alpha$ and $\beta$ are coins, though they can be any asset

- To satisfy this equation, the marginal price of asset $\beta$ with respect to $\alpha$ is always

$$m_u = \frac{R_{\beta}}{R_{\alpha}}$$
People are using these markets!

5. Uniswap Ethereum DEXes $52.1M 0.7%

52.1M USD as of 11 AM yesterday (defipulse.com)

- Celo, e.g., uses it as a price oracle
- So certainly worth analyzing!
Outline

The pricing oracle problem

Automated Market Makers

An analysis of Uniswap

Acknowledgements
Constant product markets

How do we formalize the idea that the reported price, $m_u$, must be close to the true market price, $m_p$?
Constant product markets

- How do we formalize the idea that the reported price, $m_u$, must be close to the true market price, $m_p$?

- Set up a game!
Constant product markets

- How do we formalize the idea that the reported price, $m_u$, must be close to the true market price, $m_p$?

- Set up a game!

- An arbitrageur borrows an arbitrary amount of coin $\alpha$ or $\beta$ but must pay it all back after their transaction (sound familiar?)

- The agent can then trade between two markets:
  1. Uniswap
  2. Some (infinitely liquid) reference market, with price $m_p$
Constant product markets

- How do we formalize the idea that the reported price, $m_u$, must be close to the true market price, $m_p$?

- Set up a game!

- An *arbitrageur* borrows an arbitrary amount of coin $\alpha$ or $\beta$ but must pay it all back after their transaction (sound familiar?)

- The agent can then trade between two markets:
  1. Uniswap
  2. Some (infinitely liquid) reference market, with price $m_p$

- Optimal strategy?
The arbitrage game

- Equivalent to the optimization problem

\[
\text{maximize} \quad m_p \Delta_\alpha - \Delta_\beta \\
\text{subject to} \quad (R_\alpha - \Delta_\alpha)(R_\beta + \Delta_\beta) = k.
\]

Here, $\Delta_\alpha$ is the amount of $\alpha$ traded and $\Delta_\beta$ is the amount of $\beta$ traded.

- Optimal trade $(\Delta^*_\alpha, \Delta^*_\beta)$ always satisfies:

\[
\frac{R_\beta + \Delta^*_\beta}{R_\alpha - \Delta^*_\alpha} = m_p,
\]

i.e., the new price equals to the market price!
More questions

- This game lets us ask one more important question
- Faced with these arbitrageurs, how much does manipulation cost?
More questions

- This game lets us ask one more important question

- Faced with these arbitrageurs, how much does manipulation cost?

- It is not hard to give a *per block* lower bound. Manipulating price to \((1 + \varepsilon)m_P\) costs at least

\[
C(\varepsilon) \geq KR_\alpha \min\{\varepsilon^2, \sqrt{\varepsilon}\},
\]

and \(K > 0\) a universal constant
Important points

- Lower bound is zero if manipulation is within one transaction
- Manipulation over the short term is cheap
- As is manipulation where $\varepsilon$ is small
Even more properties

- As expected, trading a fixed amount of desired coin will be cheaper as the reserves grow.
- But liquidity providers will only (rationally) add coin to reserves if they believe $m_p$ is driftless.
- Additionally, Uniswap can never be drained of coin (i.e., $R_{\alpha} + R_{\beta} \geq 2\sqrt{k}$ is always satisfied.)
Simulations confirm these results:

Price in the case of no traders (with optimal arbitrage)

An analysis of Uniswap
Simulations confirm these results:

Price in the case of trading noise

An analysis of Uniswap
Simulations confirm these results:

Initial LP utility vs HODL
Outline

The pricing oracle problem

Automated Market Makers

An analysis of Uniswap

Acknowledgements
Acknowledgements

- Dan Robinson (Paradigm)
- Hayden Adams (Uniswap)
- Regina Cai (UMA)
- Alex Evans (Placeholder)
- Jonathan Tuck (Stanford)
- Reviewers 1 and 2 (really!)
- Siddharth Seethepalli (Dover)