DeFi: data-driven characterisation of Uniswap v3 ecosystem & an *ideal crypto law* for liquidity pools

Deborah Miori and Mihai Cucuringu

Mathematical Institute, University of Oxford
Oxford-Man Institute of Quantitative Finance

22 September 2023

Agenda

Fundamentals

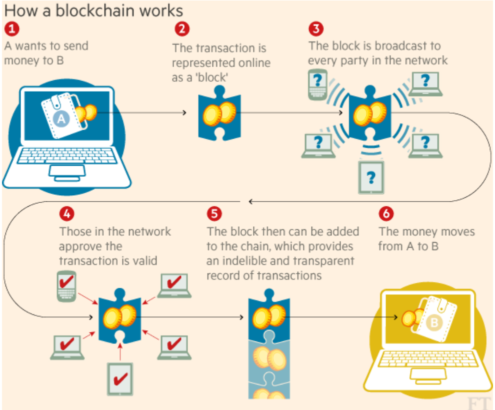


Image: Financial Times.

Famous Blockchains

- **Bitcoin** (2009) $\rightarrow$ most famous blockchain indeed, but...

- **Ethereum** (2013) $\rightarrow$ supports smart contract functionality.
  Can offer financial services without intermediaries (e.g. banks).

  Allowed the first formulation of Decentralised Finance (DeFi)
  $\rightarrow$ our focus is Uniswap (Decentralised Exchange DEX).

Intro to Uniswap

Liquidity Pool exchanging tokens X, Y - no LOB

- The **pool holds reserves** x, y of the two assets.
- Swap operation $\to$ a trader (Liquidity Taker LT) exchanges some amount Δy of asset Y for Δx of asset X against the reserves. To do this, the LT needs to pay a *fee* (feeTier γ of the pool).
- Mint/Burn operation $\to$ reserves are pooled from agents (Liquidity Providers LPs), who profit from a share of LTs fees.
- Constant Product Market Maker:

$$\begin{aligned} x \cdot y &= \kappa^2 \qquad (1) \\ &\stackrel{\text{swap}}{=} (x - \Delta x) \cdot \Big[y + (1-\gamma)\Delta y \Big]. \end{aligned}$$

Agenda

Uniswap ecosystem data

Blockchain data $\rightarrow$ fully transparent data, but complex & noisy.

Consider Uniswap data over January-June 2022:

- Minimum constraints on TVL & transactions count (txnCount)

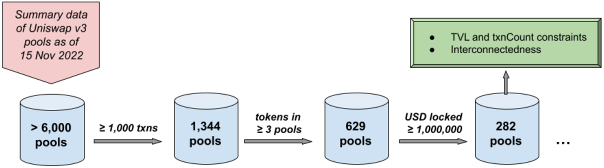


Figure: Coarse initial filtration over pools.

- Maximise the ***interconnectedness*** of final sub-universe of pools $\rightarrow$ ***weighted graph approach***... *Set of* 34 *final pools.*

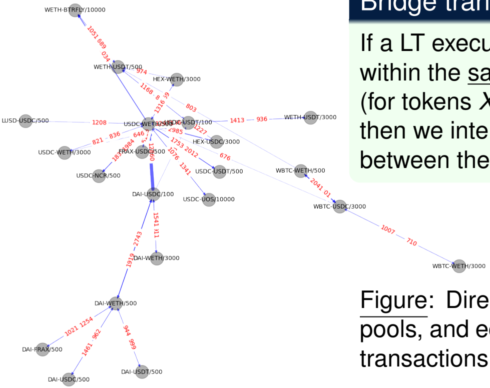
Interconnectedness

Build a **weighted graph** $G = (P, E)$, where

- node set P denotes the relevant pools
- edges $(p, q) \in E$ with $p, q \in P$ have weights w_{pq} that encode some measure of **overlap**
  1. No. of common LTs (*or* LPs) active on both pools
  2. No. of common smart contracts, called to execute LT/LP operations
  3. No. of bridge transactions between each two pools

An edge is discarded if the weight is below a threshold and the largest connected component of the graph is considered.

Interconnectedness (further)



Bridge transaction

If a LT executes 2 swaps $X \to Y$, $Y \to Z$ within the <u>same transaction</u> (for tokens X, Y, Z), then we interpret Y as a <u>bridge</u> between the action of selling X to buy Z.

Figure: Directed graph where nodes are pools, and edges are built if $\geq$ 800 bridge transactions happen between two pools.

Application: *species* of Liquidity Takers

Start: resultant **34** pools that represent the main ecosystem.

Goal: empirical investigation of LTs trading behaviour on Uniswap v3.

$\rightarrow$ Non-trivial task, e.g. since agents can easily generate numerous crypto wallets (i.e. "multiply" their identities) and are active on multiple pools.
$\rightarrow$ our six months: final set of **3, 415 LTs**.

How: we propose a novel method to express and cluster structural trading equivalence of agents on multiple environments by leveraging both network analysis and NLP techniques.

Activity of LTs

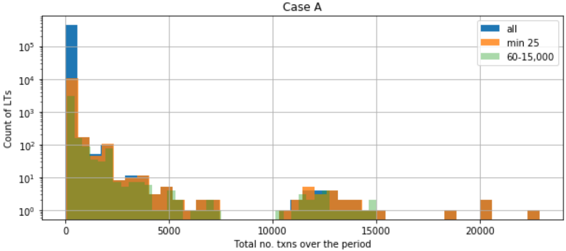
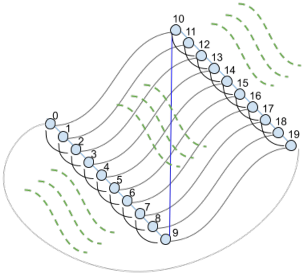


Figure: Distribution of total number of transactions (txns) performed by LTs over our 6-months window. After applying thresholds of minimum 60 and maximum $15,000$ transactions, we reach our final set of **$3,415$ LTs**.

Transaction graph

Let us introduce a **transaction graph** $G_{txn} = (S, T, W)$, characteristic of a LT.



→ **Complete weighted graph** with nodes S as the LT's swap actions.
→ Edges $(s, r) \in T$ with $s, r \in S$ have weights $w_{sr} \in W$, which encode the Δt (time difference) between the two transactions s, r.
→ Each node has a label to identify the pool that the swap was executed into.

graph2vec [1]

graph2vec $\rightarrow$ from arbitrary sized graphs $\mathcal{G} = \{G_1, G_2 ...\}$ with $G_i = (S_i, T_i, \lambda)$, to fixed-dimensions vectors. λ is a labelling function for nodes ($\sim$ words).

1. Randomly initialise the embeddings for each $G \in \mathcal{G}$
2. Extract rooted subgraphs ($\sim$ context words) to depth d, around every node s in each one of the graphs
3. For each subgraph, perform Weisfeiler-Lehman relabeling to get $g_{WL}^{d}(s)$
4. Use the subgraphs to refine the mapping f' to vectors ($\sim$ doc2vec)

Thus, our optimisation problem is

$$\max_{f'} \sum_{G \in \mathcal{G}} \sum_{s \in S} \log Pr(g_{WL}^{d}(s) | f'(G)). \tag{2}$$

Our methodology

Goal $\rightarrow$ train an embedding for each transaction graph & cluster LTs.

How $\rightarrow$ solve the optimisation problem in graph2vec [1],

$$\max_{f'} \sum_{G \in \mathcal{G}} \sum_{s \in S} \log Pr(g^{d}_{WL}(s) | f'(G)). \quad (3)$$

But need a new sampling methodology for *complete & weighted* graphs.

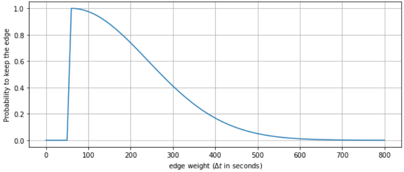
$\sim$ Keep sub-structures that focus on clusters of activity in time.
We sample neighbours such that the **probability to keep an edge** decreases with an increase in its weight, standardised with the graph's features.

$\rightarrow$ From arbitrary sized graphs $\mathcal{G} = \{G_1, G_2 ...\}$ with $G_i = (S_i, T_i, \lambda)$, to fixed-dimensions vectors. λ is a labelling function for nodes ($\sim$ words in NLP).

Cut-value

Sample neighbours such that the **probability to keep an edge** is defined by computing the *cut-value* $C(w_{sr})$ from its weight:

$$C(w_{sr}) = \frac{H(f^{scal}(w_{sr}))}{H(f^{scal}(\min W))}, \quad \text{with}$$
$$H(w_{sr}) = \sqrt{\frac{2}{\pi}} \exp\frac{-w_{sr}^2}{2}, w_{sr} \geq 0, \quad \text{and} \quad f^{scal}(w_{sr}) = \frac{w_{sr} - \min W}{(\max W)/|S|}. \tag{4}$$



$C(\min W) = 1 \rightarrow$ the shortest link(s) is chosen with probability 1.
The above function is tuned for $\min W = 60s$, $\max W = 3600s$.

Results

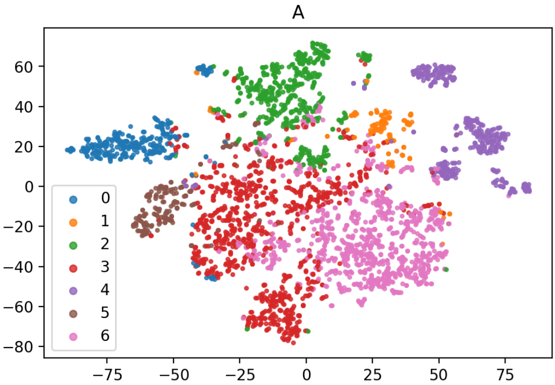


Figure: t-SNE embedding of the seven *species* of LTs unravelled.

Full characterisation of the recovered clusters

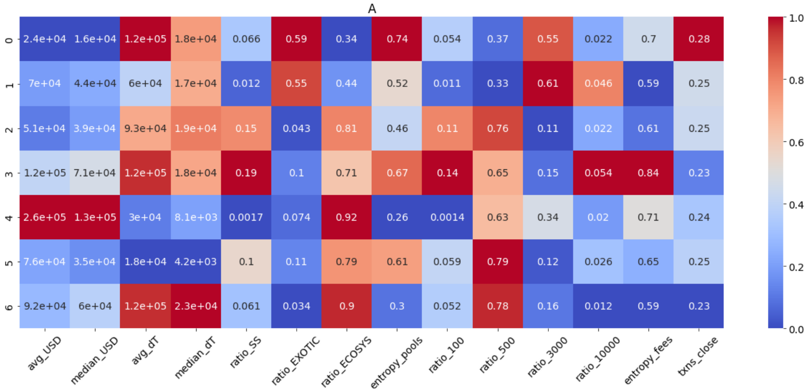


Figure: Average *external features* for the different groups of LTs unravelled over our six-months window between January-June 2022.

Interpretation

- **Groups 0 and 1** $\rightarrow$ focusing on exotic cryptocurrencies.
- **Groups 2 and 3** $\rightarrow$ advanced trading of stablecoins.
- **Group 5** $\rightarrow$ eclectic and active, maybe the smartest investors.

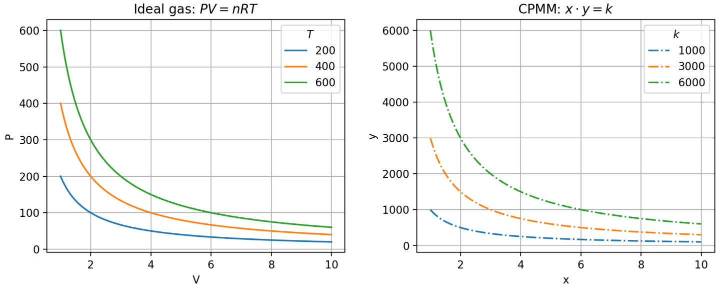
- **Groups 4 and 6** $\rightarrow$ focus on ECOSYS pools.

  Group 4 trades more USD with higher frequency, and has much higher-than-average proportion of LTs that also act as LPs ($\sim$ 16%). This group might represent more professional investors.

  Group 6 trades less money and waits longer, mainly using pools with low feeTier 500. It is the largest in size and might represent cautious retail traders that invest in less risky and highly well-known crypto possibilities.

Agenda

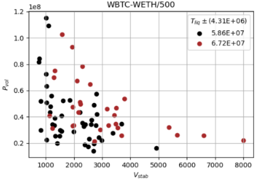
Intuition



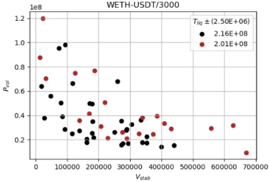
(Left) Sample of isotherms for an ideal gas, where we imposed $nR = 1$.
(Right) Trading function of a CPMM, such as Uniswap, for different levels of liquidity k...
$\rightarrow$ Temperature $\sim$ Liquidity

Question: can we try to encompass all the dynamics of the ecosystem in one formula?

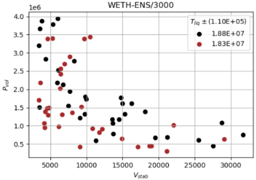
Empirical isotherms



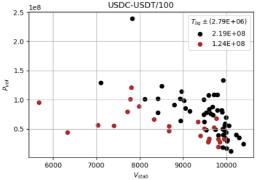
(a) WBTC-WETH/500 pool.



(b) WETH-USDT/3000 pool.



(c) WETH-ENS/3000 pool.



(d) USDC-USDT/100 pool.

Figure: From the proposed ideal crypto law, we plot related empirical isotherms on the (P_{vol}, V_{stab})-plane for a sample of pools.

Analogy

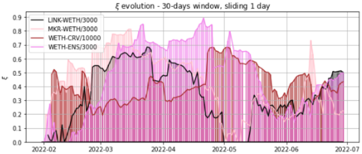
Ideal gas law		Ideal crypto law	
Symbol	Meaning	Symbol	Meaning
P	pressure	P_{vol}	daily USD volume traded by LTs
V	volume	$V_{stab} = \text{STD}(Z)^{-1}$	daily stability of the exchange rate Z
n	moles of particles	$n_{fee} = feeTier^{-1}$	stimulus to LTs' activation
R	gas constant	R_{pool}	pool crypto constant
T	temperature	T_{liq}	daily liquidity, i.e. proxyTVL value

We propose an analogy in which each liquidity pool is a gas. Thus,

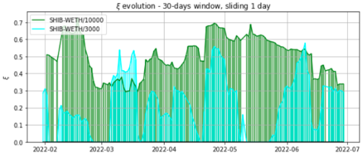
$$\begin{aligned} P_{vol} \cdot V_{stab} &= n_{fee} \cdot R_{pool} \cdot T_{liq} \\ \Rightarrow P_{vol} \cdot \text{STD}(Z)^{-1} &= feeTier^{-1} \cdot R_{pool} \cdot T_{liq}, \end{aligned} \tag{5}$$

Data to estimate the coefficient of this regression $\rightarrow$ R^2 is our *cryptoness* ξ.
"Pools with high cryptoness adhere well to our ideal crypto law model".

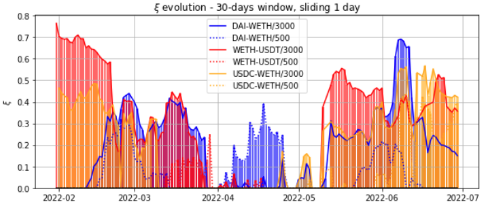
Results



(a) Evolution of ξ for a sample of pools exchanging exotic tokens.



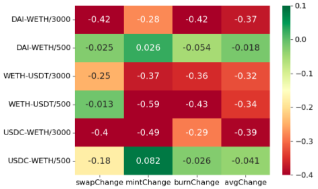
(b) Evolution of ξ for two pools exchanging SHIB-WETH.



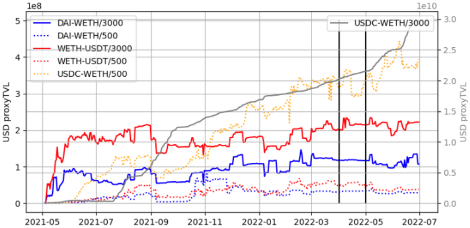
(c) Evolution of ξ for a sample of pools exchanging WETH vs. stablecoin. The dominance of different pools over time can lead to hypothesise cryptoness as a measure of the varying health of pools and indicator of where market participants should prefer to be active on.

Figure: Evolution of cryptoness ξ for different subsets of pools, where ξ is computed from the regression over a 30-days window of instances, sliding one day each time.

Cryptoness as *health* indicator of Liquidity Pools



(a) Variation in activity, i.e. in the frequency of the different transaction types, over April.



(b) Our proxyTVL in USD.

Figure: Variation of activity and evolution of liquidity between our six pools exchanging WETH against a stablecoin, over the period January-June 2022.

Note: The DAI-WETH/500 pool is the only venue with some cryptoness in April 2022. And indeed, it reports both the least damage in activity and no drop in TVL that month.

Use cases: of potential interest to practitioners&regulators to develop monitoring tools.

Agenda

Conclusions

Blockchain-based ecosystems are quite recent concepts.
Nevertheless, there is already a very interesting amount of data available.

Our contributions are:

- We focus on Uniswap and introduce a **workflow** to define a sensible subset of relevant liquidity pools.
- We show the existence of **seven "species" (clusters) of LTs** that exhibit interpretable characteristics.
- Finally, we propose a novel metric to help in assessing the ***health*** of different liquidity pools.

Thank you!

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4307924

References I

- Annamalai Narayanan, Mahinthan Chandramohan, Rajasekar Venkatesan, Lihui Chen, Yang Liu, and Shantanu Jaiswal.
  graph2vec: Learning Distributed Representations of Graphs, 2017.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean.
  Efficient Estimation of Word Representations in Vector Space, 2013.
- Aditya Grover and Jure Leskovec.
  node2vec: Scalable Feature Learning for Networks, 2016.
- Quoc V. Le and Tomas Mikolov.
  Distributed Representations of Sentences and Documents, 2014.

Uniswap Trading Example

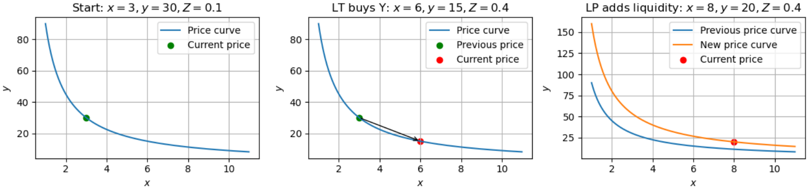


Figure: From left to right, possible evolution of a liquidity pool.

1. Initial state of the pool
2. State after a swap operation occurs (done by a LT)
3. State after further provision of liquidity (done by a LP)

Final Pools Case *A*

The final list of 34 pools that we propose to consider for LTs analyses is:

DAI-WETH/3000, CEL-WETH/3000, USDC-UOS/10000, DAI-USDC/100, SPELL-WETH/3000, WETH-CRV/10000, USDC-USDT/500, DAI-FRAX/500, WETH-BTRFLY/10000, GALA-WETH/3000, WETH-USDT/3000, WBTC-USDC/3000, DAI-USDT/500, UNI-WETH/3000, WETH-ENS/3000, DAI-USDC/500, WBTC-WETH/500, MATIC-WETH/3000, DAI-WETH/500, WETH-USDT/500, USDC-WETH/500, LINK-WETH/3000, WBTC-WETH/3000, FXS-WETH/10000, FRAX-USDC/500, USDC-WETH/3000, USDC-WETH/10000, LUSD-USDC/500, HEX-USDC/3000, USDC-NCR/500, SHIB-WETH/3000, DYDX-WETH/3000, USDC-USDT/100, HEX-WETH/3000.

Regarding LP pools, we have instead the following 19 pools:

WETH-CRV/10000, MKR-WETH/3000, WETH-USDT/3000, WBTC-USDC/3000, UNI-WETH/3000, WETH-ENS/3000, WBTC-WETH/500, MATIC-WETH/3000, DAI-WETH/500, WETH-USDT/500, USDC-WETH/500, LINK-WETH/3000, WBTC-WETH/3000, USDC-WETH/3000, SHIB-WETH/3000, WBTC-USDT/3000, USDC-USDT/100, USDC-USDT/500, SHIB-WETH/10000.

Natural Language Processing (NLP)

word2vec word embedding technique [2]
→ Consider sentences as directed subgraphs with nodes as words
→ A shallow neural network maps each word to a unique vector, capturing syntactic and semantic regularities.

$$\left(x_{apple} - x_{apples} \approx x_{car} - x_{cars}\right),$$
$$\left(x_{king} - x_{man} + x_{woman} \approx x_{queen}\right)$$

Extensions:

node2vec [3] → Map nodes in a graph to vectors by maximising the likelihood of preserving network neighbourhoods (at various depths) of nodes.

doc2vec [4]

....

NLP Extensions (1)

node2vec [3]:
it maps nodes in a graph to vectors by maximising the likelihood of preserving network neighbourhoods of nodes, i.e.

$$\max_f \sum_{s \in S} \log Pr(N_L(s)|f(s)), \tag{6}$$

where

- $G = (S, T)$ is a graph with nodes S and edges T,
- f is the mapping function for nodes to n-dimensional vectors to learn,
- and $N_L(s) \subset S$ is the network neighbourhood of node s generated with
- sampling strategy $L \rightarrow$ biased random walk procedure, that either focuses on sampling a broader set of immediate neighbours, or a sequence of deeper nodes at increasing distances.

NLP Extensions (2)

doc2vec [4]:
it learns continuous fixed-length vector embeddings from variable-length pieces of text, i.e. sentences, paragraphs and documents.

graph2vec [1]:
from arbitrary sized graphs $\mathcal{G} = \{G_1, G_2 ...\}$ with $G_i = (S_i, T_i, \lambda)$, to fixed-dimensions vectors $\rightarrow \lambda$ is a labelling function for nodes ($\sim$ words).

External Features

To assess our clustering, we compute the following external features for each individual cluster:

- average and median USD traded,
- average and median time Δt in seconds between transactions,
- proportion of transactions performed in “SS”, “EXOTIC” or “ECOSYS” pools, and related entropy,
- proportion of transactions performed in pools with a specific feeTier, and related entropy,
- proportions of trades on days when the SP LargeCap Crypto Index[1] increased or decreased in value, or when the market was closed.

[1] https://www.spglobal.com/spdji/en/indices/digital-assets/sp-cryptocurrency-largecap-index/#overview