Dynamic Automated Market Makers for Decentralized Cryptocurrency Exchange

Bhaskar Krishnamachari, Qi Feng, Eugenio Grippo
Viterbi School of Engineering, University of Southern California
Los Angeles, CA 90089, USA
{bkrishna, fengqi, egrippo}@usc.edu,

2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) | 978-1-6654-3578-9/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICBC51069.2021.9461100

***Abstract*—Decentralized cryptocurrency exchange protocols such as Uniswap, Curve and other types of Automated Market Makers (AMMs) maintain a liquidity pool (LP) of two or more assets constrained to maintain at all times a mathematical relation to each other, defined by a given function or curve. We propose a dynamic AMM approach where input from a market price oracle is used to modify the mathematical relationship between the assets so that the pool price continuously and automatically adjusts to be identical to the market price. This approach eliminates arbitrage opportunities.**

I. Introduction

Curve-based Automated Market Makers (AMM) were recently introduced to address the challenges in a currency exchange context. They are currently one of the areas of decentralized finance receiving the most attention. Instead of relying on the traditional market makers to provide liquidity, decentralized exchanges utilizing curve-based AMMs, such as Bancor [1], Uniswap [2], StableSwap/Curve [3] and many others implement a liquidity pool (LP) using smart contracts on a blockchain. In this model, liquidity providers supply single or multiple types of tokens to the designated liquidity pools, and traders exchange against the pools of tokens instead of relying on order matching. The liquidity pool of these AMMs track a pre-defined mathematical function (curve), thus determining how many tokens of one type to provide to a trader in exchange for a certain amount of another. Curve-based AMMs provide a continuous supply of liquidity compared to the order book model. Additionally, depending on the mathematical function (curve) utilized, they can potentially allow for a wide range of exchange prices. However, the token price within a liquidity pool for a given AMM (which we refer to as the pool price) might be different from the market price.

When such a gap occurs on a decentralized AMM-based exchange, arbitrageurs may have the opportunity to buy or sell tokens to set the pool price equal to the market price, restoring equilibrium. However, in some cases, particularly when the market price changes dramatically, the AMM-based LP could lose liquidity with respect to one or more of the assets. We propose in this work a new dynamic AMM approach. It utilizes input from a market price oracle to modify the mathematical relationship between the assets so that the pool price continuously and automatically adjusts to be identical to the market price. This eliminates arbitrage opportunities and helps the AMM-based LP maintain liquidity and total value over a wide range of market prices.

978-1-6654-3578-9/21/$31.00 ©2021 IEEE

II. AMM Curves

Consider a liquidity pool with two coins, whose amounts are denoted by x and y. For convenience, we will refer to these two tokens as X and Y. The AMM will allow the exchange of one token for another following a given function $y = f(x)$.

We refer to a plot of this function showing all allowed combinations of y and x as the *AMM curve*. For example, there can be a constant product curve, which is $y = \frac{k}{x}$ or a constant sum curve, which would be denoted as $y = c - x$. It is generally considered reasonable for the AMM curve to be convex and monotonically decreasing because this ensures (as we shall see in the next section) that the price for the token X is monotonically decreasing as a function of its availability in the pool, as should be expected of a typical supply curve.

Given an AMM curve, we can derive the price of the X token as follows:

$$p_X(x, y) = -\frac{dy}{dx} \tag{1}$$

For example, for the constant product curve, we would get $p_X(x, y) = \frac{k}{x^2}$ and likewise for the constant sum curve, we would get that $p_X(x, y) = 1$. A plot of $p_X(x, y)$ versus x shows how the price of token X varies with its supply in the liquidity pool. Such a curve is referred to as a *price curve*.

III. Dynamic AMM

In the prior work on AMMs, the curve has a fixed form and the exact shape is determined by the initial total liquidity. E.g., in the constant product curve, the parameter $k = x_i \cdot y_i$ where x_i, y_i are the initial amounts of the two tokens. In other words, the curve can only change if the liquidity providers add/remove tokens from the pool, but not from trading activity.

Consider a trade that happens while the market price of token X remains unchanged at some price p_{mkt}. If the pool changes from the state (x_o, y_o) to a new state (x_n, y_n), then the pool price would potentially change from $p(x_o, y_o)$ to $p(x_n, y_n)$ (assuming the curve is not the constant-sum curve in which case there is no change in the pool price). This can result in at least a temporary difference between the pool price and the market price. As we do in the rest of the paper, we are assuming here that the pool's capitalization is a relatively small fraction of the total market capitalization of the underlying assets so that the market price is not determined or affected by the pool price.

Another reason for a temporary difference between the pool price and the market price could be that the market price changes due to some external market conditions. In either case, traditionally, it is expected that these temporary differences will be erased by the action of arbitrageurs, restoring the pool price back to the market price.

We propose a new mechanism that instead changes the curve every time the market price changes in such a way as to ensure that the current pool price will always equal the market price, *without requiring action by external arbitrageurs*. We illustrate below how this new mechanism would generalize the constant-product and constant-sum curves – the same approach can be used to generalize other smooth, decreasing, convex curves to the dynamic setting as well.

A. Dynamic curve adjustment to generalize constant-sum

In this case, we can describe the market-price-tracking dynamic curve as follows:

$$p_{mkt}(t) \cdot (x(t) - a(t)) + y(t) = c \tag{2}$$

Here, the parameter $a(t)$ will also be adjusted dynamically when the market price changes, to ensure that the new linear curve passes through the current pair of $(x(t), y(t))$ values. For simplicity, say the market is initialized at some pair $(x(0), y(0)$ at a market price of 1. Then c could be set to be $x(0) + y(0)$, with the original $a(0) = 0$.

If the market shifts to a price of $p_{mkt}(t)$ at some time t and the liquidity pool at this arbitrary time is $(x(t), y(t))$, then the value of $a(t)$ will also be adjusted as follows to match the above dynamic curve:

$$a(t) = x(t) - \frac{c - y(t)}{p_{mkt}(t)} \tag{3}$$

Intuitively, this dynamic curve is always a line that has the slope corresponding to the current market price and always passing through the current liqudity pair $(x(t), y(t))$.

Any trade that happens uses the current (instantaneous) curve. This allows the constant sum AMM to flexibly support a wider range of market prices while still providing 0 slippage compared to the original design (which allows only a fixed pool price and thus will not work when the market price is dramatically different).

B. Dynamic curve adjustment to generalize constant-product

In this case, we can describe the market-price-tracking dynamic curve as follows:

$$w(t) \cdot (x(t) - a(t)) \cdot y(t) = k \tag{4}$$

Or alternatively, as:

$$y(t) = \frac{\frac{k}{w(t)}}{x(t) - a(t)} \tag{5}$$

Note that in the above expressions, $x(t)$ and $y(t)$ must always be strictly positive; $a(t)$ must be constrained to be always strictly less than $x(t)$; and $w(t)$ should always be strictly positive. The instantaneous price corresponding to the dynamic version of the constant product curve can be defined as follows:

$$p_X(t) = \frac{k}{w(t)} \cdot \frac{1}{(x - a(t))^2} \tag{6}$$

When the market price changes, then both $w(t)$ and $a(t)$ will have to be changed in order to (a) make sure that the new market price $p_{mkt}(t)$ matches $p_X(t)$ in equation (6) and (b) x(t),y(t) match the curve described in equation (4). Thus we have to solve two equations and two unknowns. The solution turns out to be the following :

$$\begin{aligned} a(t) &= x(t) - \frac{y(t)}{p_{mkt}(t)} \\ w(t) &= \frac{k \cdot p_{mkt}(t)}{y(t)^2} \end{aligned} \tag{7}$$

We remark: the first expression above ensures the requirement mentioned above that $a(t)$ will remain strictly less than $x(t)$ and the second expression ensures that $w(t)$ is strictly positive, so long as k, $p_{mkt}(t)$, $x(t)$ and $y(t)$ are all kept strictly positive at all.

IV. Conclusions

We have introduced a new approach to dynamic curve-based AMM decentralized exchanges that utilizes an oracle with a real-time market price feed to continuously and automatically adjust the pool price to the market price. In such a dynamic AMM, there is no room for arbitrage, benefiting liquidity providers (for more details, see [4]).

From a practical perspective, implementing such a dynamic AMM requires the use of a low-latency and accurate market price oracle. It would be of great interest to develop and evaluate a real-world decentralized exchange based on this approach.

References

[1] E. Hertzog, G. Benartzi, and G. Benartzi, "Bancor protocol:continuous liquidity for cryptographic tokens through their smart contracts," tech. rep., 2018.

[2] H. Adams, N. Zinsmeister, and D. Robinson, "Uniswap v2 core," tech. rep., 2020.

[3] M. Egorov, "Stableswap-efficient mechanism for stablecoin liquidity," tech. rep., 2019.

[4] B. Krishnamachari, Q. Feng, and E. Grippo, "Dynamic curves for decentralized autonomous cryptocurrency exchanges," *arXiv preprint arXiv:2101.02778*, 2021.