Analysis of the RMM-01 Market Maker

Waylon Jepsen
Primitive Bits & Colorado State University
Fort Collins, Colorado, USA
jepsen@primitive.xyz

Colin Roberts
Primitive Bits
Fort Collins, Colorado, USA
colin@primitive.xyz

Abstract

Constant Function Market Makers (CFMMs) are a popular market design for Decentralized Exchanges (DEXs). Liquidity Providers (LPs) supply the CFMM with assets to enable trades. In exchange for providing this liquidity, an LP receives a token that replicates a payoff determined by the trading function used by the CFMM. In this paper, we study a time-dependent CFMM called RMM-01. The trading function for RMM-01 is chosen such that LPs recover the payoff of a Black–Scholes priced covered call. First, we introduce the general framework for CFMMs. After, we analyze the pricing properties of RMM-01. This includes the cost of price manipulation and the corresponding implications on arbitrage. Our first primary contribution is from examining the time-varying price properties of RMM-01 and determining parameter bounds when RMM-01 has a more stable price than Uniswap V2. Finally, we discuss combining lending protocols with RMM-01 to achieve other option payoffs which is our other primary contribution.

ACM Reference Format:
Waylon Jepsen and Colin Roberts. 2022. Analysis of the RMM-01 Market Maker. In *Proceedings of ACM CCS Workshop on Decentralized Finance and Security (ACM DeFi '22).* ACM, New York, NY, USA, 9 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Blockchains are distributed computing environments that enable the construction of permissionless censorship-resistant systems such as Decentralized Exchanges (DEXs) [13, 17, 21, 27]. Decentralized financial instruments mitigate undesirable power distributions in existing traditional financial markets [16]. Currently, there are billions of dollars worth of value in Decentralized Finance (DeFi) protocols [1].

DEXs are programs that run in a decentralized computing environment that automate the function of market makers in traditional financial exchanges [10]. Distributed computation has a monetary cost quantified on the Ethereum network in units of *gas*. This constraint provides incentive for simpler market design. One such design is a Constant Function Market Maker (CFMM) which is characterized by it's unique trading function [3, 4]. The trading function defines which trades are valid and assigns a price based on the current supply of reserves.

It was proven in [7] that for any concave, non-negative, non-decreasing payoff function, there exists a corresponding CFMM replicating that payoff to Liquidity Providers (LPs). These correspondents are called Replicating Market Makers (RMMs). One such example of an RMM is the Black–Scholes covered call RMM, currently implemented as RMM-01 by Primitive [22]. This work will analyze RMM-01.

In section 2, we introduce a formal framework for CFMMs and examine the trading functions of Uniswap V2 and RMM-01. Much of this background summarizes work done in [3]. We present our main results in Section 3. In Section 3.1, we compute the price impact of a swap. Subsequently, in Section 3.2 we derive the cost of price manipulation and examine the implications on arbitrage. In Section 3.3, we evaluate and identify parameter bounds where RMM-01 has less price impact than Uniswap V2. Lastly, in Section 3.4, we examine a methodology for composing the RMM-01 LP tokens with other DeFi mechanisms to achieve the payoffs of other Black–Scholes priced options.

2 Background

First, we will define a CFMM and show how the trading function is used to determine valid trades. This leads to discussion of LPs and how they earn rewards. Since order books are a familiar concept in finance, we describe a CFMMs that acts as an order book as a means of comparison. Options are another familiar concept which we introduce briefly before defining the CFMM called RMM-01 that allows LPs to achieve a payoff of a Black–Scholes Covered Call (CC).

ACM DeFi '22, November 11, 2022, Los Angeles, CA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

[1]At the time of writing, the total value locked in Uniswap V2 comes to $6.77 billion.

2.1 Constant Function Market Makers

In Traditional Finance (TradFi), market makers facilitate the buy-and-sell orders that make up an order book and the two key actors are buyers and sellers. In DeFi, trades are facilitated against the reserves of assets of a CFMMs. For example, suppose we have a collection of n assets (e.g., tokens) that can be exchanged for another. The reserves of n assets $\boldsymbol{R} \in \mathbb{R}_+^n$ is called an *n-asset liquidity pool* where for every $i \in \{1, \ldots, n\}$, the quantity R_i represents the quantity of asset i in the pool. Reserves change when trades are executed.

Let $\boldsymbol{\Delta}, \boldsymbol{\Lambda} \in \mathbb{R}_+^n$ be the *tendered basket* and *received basket*, respectively. We refer to the tuple, $(\boldsymbol{\Delta}, \boldsymbol{\Lambda}) \in \mathbb{R}_+^n \times \mathbb{R}_+^n$, as a *proposed trade*. Specifically, Δ_i and Λ_i denote the amount of asset i tendered. Fees are typically applied to the tendered basket and they are a means of incentivizing users to provide liquidity. We define the *fee* as a parameter $\gamma \in (0, 1]$.

Definition 2.1. A *Constant Function Market Maker (CFMM)* is an n-asset pool $\mathbb{R}_+^n$, and a *trading function* φ

$$\varphi\colon \mathbb{R}_+^n \to \mathbb{R}. \tag{1}$$

Given any $\mathbf{R}$, the value $\varphi(\mathbf{R}) = k$ is called the *invariant*.

CFMMs have two independent actors. Actors who tender a basket of assets $\boldsymbol{\Delta}$ in order to receive a nonzero basket $\boldsymbol{\Lambda}$ are *swappers*. Actors who provide those assets that can be swapped are called the *LPs*. Let us first discuss swappers.

Definition 2.2. Let $(\Delta, \Lambda) \neq 0$, then this trade is a *valid swap* if

$$\varphi(\boldsymbol{R} + \gamma\Delta - \Lambda) = \varphi(\boldsymbol{R}). \tag{2}$$

Graphically, definition 2.2 dictates that valid swaps move reserves along the *invariant curves* where $\varphi = k$. An illustration of these curves is given in Figure 3 for both Uniswap V2 and RMM-01. If $\gamma = 1$, we see that a swapper is simply paying $\boldsymbol{\Delta}$ in order to receive $\boldsymbol{\Lambda}$. In the case that $\gamma < 1$ (which is typical), the trade is accepted based on the discounted tendered basket, $\gamma\boldsymbol{\Delta}$, but the reserves are still increased by the full $\boldsymbol{\Delta}$. This remainder $(1-\gamma)\boldsymbol{\Delta}$ serves to increase the value the LP's share of the pool which we call a Liquidity Provider Token (LPT). We discuss LPs further in Section 2.3.

Suppose that we have only two assets: Token1 and Token2. In TradFi, an order book will provide the last *market price* for which a trade was executed. This can be thought of as the exchange ratio $S = \frac{\Lambda_1}{\Delta_2}$. Here, we will assume that asset $i = n$ is the numeraire and that it is some stable reference such as USDC.

Definition 2.3. Given a CFMM the *price vector* is

$$\boldsymbol{P} \coloneqq \nabla\varphi, \tag{3}$$

the *reported price of asset i* is

$$p_i \coloneqq \frac{P_i}{P_n}, \tag{4}$$

and the *value of the reserves* is

$$V \coloneqq \frac{1}{P_n}\boldsymbol{P}^T\boldsymbol{R}. \tag{5}$$

Note that since φ is a function of the reserves, so is the reported price p. In [4, Section 2.4] the authors compute examples of pricing with two DEXs: Balancer and Curve [5, 17]. Uniswap V2 has the trading function $\varphi_{\text{Uni}}(\boldsymbol{R}) \coloneqq R_1R_2$. The associated price is the ratio of the reserves $p_1 = \frac{R_2}{R_1}$. For 2-asset pools we will put $p \coloneqq p_1$. For more on Uniswap, see [9].

When a swap occurs on a CFMM, the reserves change. If the invariant curve for a CFMMs has nonzero curvature, the change in reserves necessarily results in a change in price. For a 2-asset pool, we can define the following.

Definition 2.4. Fix reserves $\boldsymbol{R} = (R_1, R_2)$ and (without loss of generality[2]) a valid swap Δ_1 for Λ_2, then the *price impact due to* Δ_1 is

$$g(\Delta_1) = \frac{d\Lambda_2}{d\Delta_1}. \tag{6}$$

The above definition appears in [6] and more recently in [19]. For example, take φ_{Uni} and suppose that $\Delta_1 > 0$, $\Delta_2 = 0$, and that we require $\Lambda_1 = 0$. This is reasonable since round trip trades are never profitable [3, Section 2.5]. Following Definition 2.2, a valid swap for φ_{Uni} satisfies

$$\varphi_{\text{Uni}}(\boldsymbol{R}) = R_1R_2 = (R_1+\gamma\Delta_1)(R_2-\Lambda_2) = \varphi_{\text{Uni}}(\boldsymbol{R}+\gamma\boldsymbol{\Delta}-\boldsymbol{\Lambda}) \tag{7}$$

which implies

$$\Lambda_2 = \frac{\Delta_1 R_2}{R_1 + \Delta_1}. \tag{8}$$

Given that the price $p = \frac{R_2}{R_1}$, we have that the price impact due to Δ_1 is

$$g(\Delta_1) = \frac{\gamma R_1 R_2}{(R_1 + \gamma\Delta_1)^2}. \tag{9}$$

One can define a price impact for arbitrary n-asset CFMMs, but note that there is not always a unique received basket $\boldsymbol{\Lambda}$ for a given tendered basket $\boldsymbol{\Delta}$.

It has been shown that depending on a price oracle can introduce centralization risk. If a price oracle is manipulated or malfunctions in any way, the financial consequences can be drastic [25]. Note that the results of [4] show that arbitrageurs increase market efficiency and can be utilized to mitigate oracle dependencies.

2.2 Order Book Comparison

In TradFi, a common market architecture is an order book. In the order book design, buyers and sellers provide their liquidity in the form of limit orders at a specific price. Here we will examine the collection of CFMMs that describe an order book and see why they have not been used in the design of decentralized exchanges on the Ethereum network.

[2] You can achieve a swap Δ_2 for Λ_1 just by change of sign when there are no fees.

An order book can be built as a collection of 2-asset CFMMs with a trading function called a *constant sum trading function.* We define a generic constant sum trading function by

$$\varphi_{\text{cs}}(\mathbf{R}) = P_1 R_1 + P_2 R_2 = \boldsymbol{P}^T \boldsymbol{R}, \tag{10}$$

such that $P_1, P_2 \in \mathbb{R}_+$. The constant sum trading function dictates that valid swaps only execute at a single reported price which is akin to a limit order. The constants P_1 and P_2 define the price by

$$p_{\text{cs}} = \frac{\partial_1 \varphi_{\text{cs}}}{\partial_2 \varphi_{\text{cs}}} = \frac{P_1}{P_2} \tag{11}$$

where ∂_i is the partial derivative with respect to the i^{th} reserve. A visual is provided in Figure 2.

When building applications on the blockchain, there is a monetary cost of computation. Thus, an on-chain order book becomes prohibitively expensive since it requires an enormous collection of constant sum markets. Instead, a CFMM allows for a continuous range of prices using a single continuous nonlinear trading function. The dominant market architecture for DEXs is nonlinear CFMMs.

2.3 Liquidity Providers

LPs tender assets to a pool in exchange they receive a quantity of LPTs representing their share of pool ownership. The LPT can always be exchanged for reserve assets if an LP wishes to exit their position. Upon exiting, an LP receives a basket $\boldsymbol{\Lambda}$ for their LPT. Their share is defined by their initial tendered basket $\boldsymbol{\Delta}$ and the accumulated swap fees. For more detail, see [3]. We refer to the actions an LP performs as *liquidity changes.*

Swappers and LPs have a symbiotic relationship so long as $\gamma \neq 1$. For any swap $(\boldsymbol{\Delta}, \boldsymbol{\Lambda})$ there will be $(1-\gamma)\boldsymbol{\Delta}$ placed into the pool. Hence, swaps increase the amount of the underlying tokens that an LPT is worth. More detail about the change in value for an LPT is given in [3, Section 2.6].

In CFMMs, the reported price is updated through arbitrage. To capitalize on an arbitrage opportunity, the arbitrageurs must execute swaps on two exchanges and each swap executed by the arbitrageur on a CFMM generates fee paid to LPs. This mechanism drives RMM-01 to replicate the payoff of a CC. We do not go into the exact detail here, but instead we refer the reader to the original papers [8, 21] that describes this trading function and its implementation.

A LP executes a trade of the form $(\boldsymbol{\Delta}, 0)$ or $(0, \boldsymbol{\Lambda})$ that modifies the invariant k but not the reported price[3]. Specifically, LPs must provide or remove liquidity along directions that preserve the gradient of the trading function.

[3]LPs and swappers are orthogonal actors in a CFMM. The fee γ can be considered as a "rotation" that converts the impact of a swapper to an LP geometrically.

Definition 2.5. A *valid liquidity change* is a trade $(\boldsymbol{\Delta}, 0)$ or $(0, \boldsymbol{\Lambda})$ such that

$$p(\boldsymbol{R}) = p(\boldsymbol{R} + \boldsymbol{\Delta}) = p(\boldsymbol{R} - \boldsymbol{\Lambda}) \tag{12}$$

The liquidity changes occur along the *constant price curves* that can be seen in Figure 3 for both Uniswap V2 and RMM-01.

2.4 Black-Scholes Option Pricing

The price of a Black–Scholes option depends on the chosen *strike price*, K, and the time until expiration τ [11]. Once these parameters are chosen, the option's price changes due to changes in market price S of the underlying asset, the *implied volatility* σ of the underlying [4], and the *time to expiry* τ of the contract[5].

The value of an option can be given in terms of the variables d_1 and d_2 defined as

$$d_1(S,\tau) = \frac{\ln(\frac{S}{K}) + \frac{1}{2}\sigma^2\tau}{\sigma\sqrt{\tau}} \qquad \text{and} \qquad d_2(S,\tau) = d_1 - \sigma\sqrt{\tau}. \tag{13}$$

Note that throughout this paper, we denote the zero-mean unit-variance Gaussian probability density function (PDF), cumulative distribution function (CDF), and quantile functions by ϕ, Φ, and Φ^{-1}, respectively.

Definition 2.6. The *value of a long call* is

$$V_{\text{C}}(S,\tau) = S\Phi(d_1) - K\Phi(d_2) \tag{14}$$

and the *value of a covered call* is

$$V_{\text{CC}}(S,\tau) = S\Phi(-d_1) + K\Phi(d_2). \tag{15}$$

Black–Scholes priced options satisfy a put-call parity [18] that allows for a put to be priced immediately. The *value of a long put* is

$$V_{\text{P}}(S,\tau) = V_{\text{C}} - S + K = K\Phi(-d_2) - S\Phi(-d_1). \tag{16}$$

2.5 RMM-01

In the paper *Replicating Market Makers* [8], the authors provide an example of a CFMM that approximates the payoff of a Black–Scholes CC option. This trading function was deployed on the Ethereum main-net under the name RMM-01 [21]. At the time of writing, the average total value locked on RMM-01 is $550,965.38 and the average daily swap volume is $11,058.58 [14].

The behavior of RMM-01 is unlike options in TradFi. Specifically, it is not a means to buy or sell options contracts as it does not act as a market counterparty. An RMM-01 pool is initialized when the LP chooses a strike price K (of ETH in USDC), an implied volatility σ, the time to expiry τ, and adds liquidity to the pool.

[4]While implied volatility is variable in traditional options, it is constant for RMM-01 which results in effectively no vega.

[5]Black–Scholes assumes a risk-free interest rate [11]. We omit this feature.

Once a pool is created, other LPs are free to provide liquidity to existing pools. Swappers (commonly arbitrageurs) enticed by profit opportunities from pricing discrepancies between RMM-01 and another market. Since LPs grow their position through swap fees, arbitrage is a necessary ingredient for accurate replication of the Black–Scholes CC payoff that RMM-01 claims. As an added benefit, this methodology eliminates the dependence upon an options market counterparty.

Consider the 2-asset pool with the RMM-01 trading function given by

$$\varphi_{\text{RMM-01}} := -K\Phi(\Phi^{-1}(1-R_1)-\sigma\sqrt{\tau})+R_2. \tag{17}$$

Token1 is typically called the *risky asset* (or the *underlying* in TradFi literature). Token2 is called the *stable asset* and is our numéraire. Note that $\varphi_{\text{RMM-01}}$ is evaluated on a per-LPT basis and the invariant k is not a measure of total liquidity as in Uniswap V2. Instead, k measures how accurate the replication of the payoff of a CC is being achieved. We say that there is *perfect replication* if $k = 0$. The pool is over replicating if $k > 0$ and under-replicating if $k < 0$. We can see how the invariant curves for perfect replication change over time in Figure 4.

3 Properties of RMM-01

Here, we apply some of the definitions from Section 2 to perform analysis on the RMM-01 trading function. We first compute the price impact and find the cost of manipulation and associated arbitrage implications. Furthermore, we compare our results to Uniswap V2 and introduce a bound on $\sigma\sqrt{\tau}$ in which the trading is preferable on RMM-01. Lastly, we introduce a mechanism to construct additional Black–Scholes priced payoffs.

Equation (17) implies that $R_1 \in (0,1)$ and if we assume perfect replication it must be that $R_2 \in (0,K)$ when $\tau > 0$. Applying Definition 2.3 to the trading function in Equation (17), the reported price of Token1 in terms of Token2 for an RMM-01 pool

$$p = Ke^{\Phi^{-1}(1-R_1)\sigma\sqrt{\tau}-\frac{1}{2}\sigma^2\tau}. \tag{18}$$

An LP deposits a collection of Token1 and Token2 in proportional quantities so that the reporting price p from Equation (18) matches the market price S. For instance, suppose a user has 1 ETH (Token1) and that the market price for ETH in USDC (Token2) is $S = 1000$. Then the reserves of ETH per LPT are given by

$$R_1 = 1-\Phi\left(\frac{\ln\left(\frac{S}{K}\right)+\frac{1}{2}\sigma^2\tau}{\sigma\sqrt{\tau}}\right), \tag{19}$$

which implies that the user will deposit R_1 ETH. The user can always purchase more or fractional amounts of LPTs. Given the user's chosen K, σ, and τ, the number of USDC tokens to be deposited is

$$R_2 = K\Phi\left(\frac{\ln\left(\frac{S}{K}\right)-\frac{1}{2}\sigma^2\tau}{\sigma\sqrt{\tau}}\right). \tag{20}$$

As long as $\sigma\sqrt{\tau} > 0$, it follows that $KR_1 + R_2 < K$. This means the user may have remaining funds and can choose to purchase more LPTs.

Applying eq. (5) to eq. (17) and assuming $p = S$, we can see the value of RMM-01s LPT V_{LPT} in terms of the numeraire ($n = 2$) matches that value of a CC

$$V_{\text{LPT}} = S\Phi(-d_1)+K\Phi(d_2) = V_{\text{CC}}. \tag{21}$$

Briefly, note that when $\tau = 0$ (i.e., the pool expired), the trading function becomes a constant sum market

$$KR_1 + R_2 = K \tag{22}$$

and the reported price is

$$p|_{\tau=0} = K. \tag{23}$$

Recall that a constant sum market is equivalent to a limit order and the geometry is seen in Figure 2.

3.1 Price Impact

Since RMM-01's trading function changes over time, it follows that the price impact does as well. Also, as RMM-01 is not symmetric, we must consider price impacts of both Token1 and Token2 separately. We write $g(\Delta_1, 0)$ and $g(0, \Delta_2)$ to denote the price impacts due to tendering Δ_1 and Δ_2, respectively.

First, if the swapper tenders Δ_1 then Equation (18) implies that the new price $g(\Delta_1, 0)$ is

$$g(\Delta_1, 0) = Ke^{\Phi^{-1}(1-R_1-\gamma\Delta_1)\sigma\sqrt{\tau}-\frac{1}{2}\sigma^2\tau}. \tag{24}$$

At expiry (when $\tau = 0$), we have

$$\Lambda_2 = \frac{\gamma\Delta_1}{K}, \tag{25}$$

which implies that swappers can only exchange assets at the strike price when the pool expires. If $S > K$, then all traders are incentivized to convert the pool into Token2, so an LP recovers their LPT payoff in Token2. The opposite is also valid, so if $S < K$, the pool expires with only Token1.

If instead, a swapper tenders Δ_2, then we can find that the new price is

$$g(0, \Delta_2) = Ke^{\Phi^{-1}\left(\frac{R_2+\gamma\Delta_2}{K}\right)+\frac{1}{2}\sigma^2\tau}. \tag{26}$$

Once again, at expiry $p'|_{\tau=0} = p|_{\tau=0}$ and we find

$$\Lambda_1 = \gamma K\Delta_2. \tag{27}$$

3.2 Manipulation and Arbitrage

Suppose that we want to manipulate the reported price from $p \mapsto (1+\epsilon)p$, then we need to find a corresponding Δ_1 or Δ_2 that would lead to this price change. If the swapper wants to tender Δ_1 we use Equations (18) and (24) to see

$$\Delta_1 = \gamma^{-1}\left(1 - R_1 - \Phi\left(\Phi^{-1}(1-R_1) + \frac{\ln(1+\epsilon)}{\sigma\sqrt{\tau}}\right)\right). \quad (28)$$

If Δ_2 is tendered then we use Equations (18) and (26) to get

$$\Delta_2 = \gamma^{-1}K\Phi\left(\Phi^{-1}(1-R_1) - \sigma\sqrt{\tau} + \frac{\ln(1+\epsilon)}{\sigma\sqrt{\tau}}\right) - \gamma^{-1}R_2. \quad (29)$$

Note that Δ_1 and Δ_2 are per LPT, thus this cost increases linearly with liquidity. Arbitrageurs also need to execute a swap in order to have the reported price p match an external market price S to maximize their profits. In order for an arbitrageur to initiate a profitable swap, the reported price p must satisfy the following bounds that depend on the fee:

$$\gamma S \leq p \leq \gamma^{-1}S. \quad (30)$$

These bounds are true for all CFMMs. Given p satisfies the bounds in Equation (30), they can supply Δ_1 or Δ_2 from Equations (24) and (26) by letting $\epsilon = \gamma^{-1} - 1$ if the reported price needs to increase and $\epsilon = \gamma - 1$ if the price needs to decrease.

If the invariant curves for a 2-asset trading function have small curvature, then the price impact is also small. More detail on the relationship of curvature to capital efficiency is shown in [4, 12].

3.3 Compare with Uniswap V2

By comparing RMM-01 with Uniswap V2, we can identify explicit bounds for $\sigma\sqrt{\tau}$ such that the price impact for a swap on RMM-01 is less than on Uniswap V2. This proves to be useful for users wanting to trade large quantities of assets who don't want to be exposed to a large price tolerance commonly known as *slippage*. Existence of these bounds is due to the fact that the curvature of the invariant curves approaches zero for RMM-01 as the pool approaches expiry (see Figure fig. 4). We studied price impact in Section 3.2 and in this subsection we examine the price impact with respect to an infinitesimal swap. RMM-01 and Uniswap V2 have different trading function and consequently price assets differently. Furthermore, RMM-01 considers reserves on a per LPT basis. Suppose that RMM-01 and Uniswap V2 both report price p for Token1. How do the infinitesimal changes in price for each CFMM relate to one another? In particular, for what choices of σ and τ can RMM-01 achieve less price impact?

Recall that the reported price for Uniswap V2 is $p_{\text{uni}} = \frac{R_2}{R_1}$. For any swap tendering either asset, the reserves must be a point along the same invariant curve (see Figure 3). Hence, we can compute a directional derivative of the price along invariant curves to get an infinitesimal price impact. Assuming normalized reserves, $\boldsymbol{P}_{\text{Uni}} = \begin{pmatrix} R_2 & R_1 \end{pmatrix}^T$ which is orthogonal to the invariant curves. We can apply a rotation by $\pi/2$ via a linear transformation J to $\boldsymbol{P}_{\text{Uni}}$ yields a vector tangent to the invariant curves $J\boldsymbol{P}_{\text{Uni}} = \begin{pmatrix} -R_1 & R_2 \end{pmatrix}^T$. Computing the directional derivative along $J\boldsymbol{P}_{\text{Uni}}$ yields

$$\nabla_{J\boldsymbol{P}_{\text{Uni}}} p_{\text{uni}} = -R_1\frac{\partial p_{\text{Uni}}}{\partial R_1} + R_2\frac{\partial p_{\text{Uni}}}{\partial R_2} = 2p_{\text{Uni}}. \quad (31)$$

By the same argument for RMM-01, we get

$$\nabla_{J\boldsymbol{P}_{\text{RMM-01}}} p_{\text{RMM-01}} = \frac{p_{\text{RMM-01}}\sigma\sqrt{\tau}}{\phi(\Phi^{-1}(1-R_1))}. \quad (32)$$

If we assume $p_{\text{Uni}} = p_{\text{RMM-01}}$, then we can compare Equations (31) and (32) and determine that for the infinitesimal price impact for RMM-01 to be less than that of Uniswap at the same price, it must be that

$$\sigma\sqrt{\tau} < 2\phi(\Phi^{-1}(1-R_1)), \quad (33)$$

where R_1 is the reserves for RMM-01 given by Equation (19) and ϕ is the standard normal PDF. We can visualize eq. (33) in Figure 5.

3.4 Composability

A natural question is: given the RMM-01 LPT can replicate the payoff of a CC, can one obtain other Black–Scholes priced options from this LPT? We examine the composition of assets and mechanisms that result in the payoff of a long call $V_C(S,\tau)$ and consequently a long put. This work expands on some of the results in [23] which introduces additional composability results from RMM-01.

We propose a basic borrowing and lending mechanism that can achieve the payoff of a long call or put using RMM-01. First, we outline the approach and then we analyze the inherent risk and maximal downsides. Our results illustrate the differences between a traditional call and one built by composing borrowing with RMM-01 LPTs. This is a type of trade that can be implemented on blockchains today with no need for a new lending protocol.

In TradFi, put and call options are commonly used to mitigate risk [15] and they can play a similar role in DeFi. As of March 2022, the derivative market on centralized exchanges for cypto assets represents 62.8% of trading volume [26] expressing an apparent demand.

Assuming perfect replication of RMM-01, the value of an RMM-01 LPT is given by V_{CC} found in Equation (15). Since a CC is equivalent to a long position in the underlying and short a call, we get the relationships between V_{CC} and V_C seen in Definition 2.6. Using the same relationship, we can achieve the payoff of a call V_C by shorting a CC and adding a long position on the underlying. In terms of RMM-01, we must have a method to short the LPT and hold one unit of Token1.

To short an asset in DeFi, one can borrow the asset with collateral and sell it. This sale can be done on a DEX for either Token1 or Token2[6]. Note that the borrowed asset can be bought back at a later time or the borrowing position can be liquidated. Decentralized lending protocols also ensure at least a one-to-one loan-to-value ratio for collateral and dipping below this ratio causes liquidation. Each of these steps will be called a *transaction* (labeled **tx**) and are executed atomically. On a blockchain, collections of transactions are listed on*blocks*.

For the remainder of this section, assume that $p = S$. Let τ_{start} be the time of entering the position and τ_{end} be time of exiting the position such that $\tau_{\text{start}} > \tau_{\text{end}} \geq 0$. Suppose that a user provides Token1 (e.g., ETH) as collateral in order to borrow an LPT. The user can then immediately sell the LPT for Token1. Because of atomic execution of transactions, we can assume both transactions occur at τ_{start}. At τ_{end}, the user can buy back the LPT and repay their debt for their original collateral and exit the position unless they have already experienced liquidation. We refer to this position as a *RMM-01 synthetic call* and denote the payoff by V_{RMMC}. Thus the steps to construct the V_{RMMC} from the RMM-01 LPT are summarized as follows:

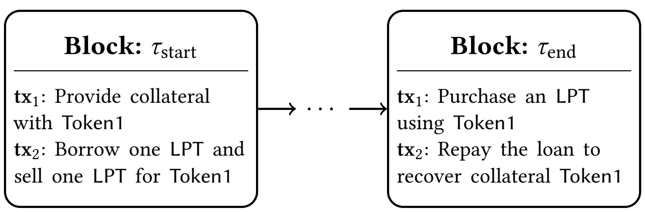


Figure 1. Entering the position of a V_{RMMC} at **Block:** τ_{start} and exiting it at block **Block:** τ_{end}.

Note that the position V_{RMMC} described above is built under the assumptions of a one-to-one loan to value ratio. In Table 1 below, the first column denotes x as the sale price of V_{LPT} in terms of Token1 at τ_{start}, the second column is the chosen collateral $y \geq x$ provided to borrow the LPT, the third column is the total amount of Token1 the user is exposed to, the fourth column is the value of V_{RMMC} given a market price S, and the final column represents the differences between a V_{RMMC} and a V_{C}.

Table 1. The positions depending on the CC parameters.

Sale	Col.	Token1	V_{RMMC}	$V_{\text{C}} - V_{\text{RMMC}}$
x	y	$x + y$	$(x + y)S - V_{\text{LPT}}$	$(x + y - 1)S$

At τ_{start} it is very possible that $x + y - 1 \neq 0$. One such case is if the user over-collateralize ($y > x$). However, the user can still achieve the long call payoff by adjusting their exposure to Token1. Specifically, the user needs to find an amount z of Token1 such that $x + y + z - 1 = 0$. The key assumptions are that there is sufficient liquidity of LPTs on a lending market and a DEX. The cost of entering the position is the collateral y equal in value to V_{LPT} at τ_{start}.

To achieve the payoff of a put, the user replaces all instances of Token1 in Figure fig. 1 with Token2. This is a consequence of put-call parity. Using eq. (16) we can see that $V_{\text{P}} = -V_{\text{CC}} + K$. In this case, the user may not achieve the correct $+K$ term need due to the value of LPT at τ_{start}.

A loan must have at least a loan-to-value ratio of one which is why it must be that $y \geq x$. The user may provide $y > x$ to ensure that their loan-to-value ratio does not dip below one even with adverse price action of Token1. In essence, the difference $y - x$ can be thought of as a choice of stop-loss. By providing $y = 1$, the user can ensure they will never face liquidation since the LPT can trade for at most one of Token1.

Suppose that the price of Token1 decreases to the point where the LPT expires containing one of Token1. This is the worst case scenario for the user long V_{RMMC}. Then, along the way, the user may find that their loan to value ratio drops below one and they will be liquidated. That is, they will necessarily forfeit their y Token1. They will, however, still keep their x Token1 and can decide what to do from there. At any rate, the maximal loss the user experiences is $yS_{\tau_{\text{start}}}$ where $S_{\tau_{\text{start}}}$ is the initial price of Token1 in terms of Token2. Once again, this implies the worst possible risk is $S_{\tau_{\text{start}}}$ if the user collateralizes using $y = 1$.

When an actor borrows assets from a DeFi protocol, they pay an interest rate on their loan for the time they are borrowing the assets. This means that there is an additional payoff to the lender and an additional cost to the borrower that we do not consider here.

4 Conclusion

In this work, we studied the time-dependent CFMM called RMM-01 which replicates the payoff of a Black–Scholes CC. Specifically, we computed the price impact due to swaps, discussed price manipulation and arbitrage, and we compared these to known bounds for another CFMM called Uniswap V2. Finally, we describe how to achieve the payoff of a long call using an external lending protocol. We do, however, have further open questions to answer.

Optimal Fees for RMM-01. It has been shown that fixed fees in the RMM-01 protocol do not yield perfect replication of a CC payoff [24]. Since RMM-01 has decreasing price impact over time, we believe that the optimal fee mechanism should also decrease over time to allow for larger swaps. Preliminary work has been done here [20].

Symmetric Asset Pools. Currently RMM-01 supports ETH and USDC but it can interact with all ERC20 tokens. One potentially interesting research question is to examine the

[6]Could be for any other token to create a more exotic position.

behavior of RMM-01 when the pool consists of two stable tokens. The trading function used by Curve [1] is often used for stable-stable swapping since there is less price impact. Does RMM-01's time-dependent replication of a CC add any benefit for a stable-stable pool? Similar to the previous thought, what are the economic implications of a pool consisting of two volatile assets such as ETH and BTC pool?

References

[1] 2022. Curve Documentation. (June 2022), 161.
[2] Hayden Adams, Noah Zinsmeister, Moody Salem, River Keefer, and Dan Robinson. 2021. Uniswap V3 Core.
[3] Guillermo Angeris, Akshay Agrawal, Alex Evans, Tarun Chitra, and Stephen Boyd. 2021. Constant Function Market Makers: Multi-Asset Trades via Convex Optimization. (July 2021), 31.
[4] Guillermo Angeris and Tarun Chitra. 2020. Improved Price Oracles: Constant Function Market Makers. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*. 80–91. https://doi.org/10.1145/3419614.3423251 arXiv:2003.10001 [math, q-fin]
[5] Guillermo Angeris, Tarun Chitra, Alex Evans, and Stephen Boyd. 2022. Optimal Routing for Constant Function Market Makers. arXiv:2204.05238 [math, q-fin]
[6] Guillermo Angeris, Alex Evans, and Tarun Chitra. [n.d.]. When Does the Tail Wag the Dog? Curvature and Market Making. ([n. d.]), 42.
[7] Guillermo Angeris, Alex Evans, and Tarun Chitra. 2021. Replicating Market Makers. *arXiv q-fin arXiv:2103.14769* (2021). arXiv:2103.14769
[8] Guillermo Angeris, Alex Evans, and Tarun Chitra. 2021. Replicating Monotonic Payoffs Without Oracles. arXiv:2111.13740 [math, q-fin]
[9] Guillermo Angeris, Hsien-Tang Kao, Rei Chiang, Charlie Noyes, and Tarun Chitra. 2021. An Analysis of Uniswap Markets. arXiv:1911.03380 [cs, math, q-fin]
[10] Massimo Bartoletti, James Hsin-yu Chiang, and Alberto Lluch-Lafuente. 2022. A Theory of Automated Market Makers in DeFi. arXiv:2102.11350 [cs]
[11] Fischer Black and Myron Scholes. 2019. The Pricing of Options and Corporate Liabilities. In *World Scientific Reference on Contingent Claims Analysis in Corporate Finance: Volume 1: Foundations of CCA and Equity Valuation*. World Scientific, 3–21.
[12] Tarun Chitra, Guillermo Angeris, and Hsien-Tang Kao. 2021. A Note on Borrowing Constant Function Market Maker Shares. (2021), 16.
[13] Michael Egorov. 2019. StableSwap - Efficient Mechanism for Stablecoin Liquidity. (2019), 6.
[14] Kim Evan. 2022. Primitivefinance/Pool-Analytics. https://github.com/primitivefinance/pool-analytics.
[15] Kenneth A. Froot, David S. Scharfstein, and Jeremy C. Stein. 1993. Risk Management: Coordinating Corporate Investment and Financing Policies. *The Journal of Finance* 48, 5 (Dec. 1993), 1629. https://doi.org/10.2307/2329062
[16] Campbell R Harvey, Ashwin Ramachandran, and Joey Santoro. 2021. *DeFi and the Future of Finance*. John Wiley & Sons.
[17] Ben Hauser. 2021. Curvefi/Curve-Contract: Vyper Contracts Used in Curve.Fi Exchange Pools. https://github.com/curvefi/curve-contract.
[18] Robert C. Klemkosky and Bruce G. Resnick. 1979. Put-Call Parity and Market Efficiency. *The Journal of Finance* 34, 5 (1979), 1141–1155. https://doi.org/10.2307/2327240
[19] Kshitij Kulkarni, Theo Diamandis, and Tarun Chitra. 2022. Towards a Theory of Maximal Extractable Value I: Constant Function Market Makers. (2022), 57.
[20] Jason Milionis, Ciamac C Moallemi, Tim Roughgarden, and Anthony Lee Zhang. 2022. Automated Market Making and Loss-Versus-Rebalancing. (2022), 21.
[21] Estelle Sterrett, Alexander Angel, and Matt Czernik. 2022. Whitepaper-Rmm-01.Pdf. https://primitive.xyz/whitepaper-rmm-01.pdf.
[22] Estelle Sterrett, Alexander Angel, Matt Czernik, and experience. 2021. Primitive Whitepaper. *PrimitiveXYZ* (2021).
[23] Estelle Sterrett, Waylon Jepsen, and Evan Kim. 2022. Replicating Portfolios: Constructing Permissionless Derivatives. https://doi.org/10.48550/arXiv.2205.09890 arXiv:2205.09890 [q-fin]
[24] The Graph. 2022. *The GraphiQL*. https://api.studio.thegraph.com/query/20803/primitive-subgraph/v0.0.2-rc0/graphql
[25] Kevin Tjiam, Rui Wang, Huanhuan Chen, and Kaitai Liang. 2021. Your Smart Contracts Are Not Secure: Investigating Arbitrageurs and Oracle Manipulators in Ethereum. In *Proceedings of the 3rd Workshop on Cyber-Security Arms Race*. 25–35.
[26] James Webb. 2022. Cryptocompare_exchange_review_2022_03_vf-2.Pdf. https://www.cryptocompare.com/media/40061772/cryptocompare_exchange_review_2022_03_vf-2.pdf.
[27] Noah Zinsmeister and Dan Robinson. 2020. Hayden Adams Hayden@uniswap.Org. (2020), 10.

A CFMM Geometry

Let us describe the geometry of CFMM trading curves briefly. First, we can see a visual for the invariant curves for different values of k in the constant sum market trading function φ_{cs} described by Equation (10). Alongside this, the price vector $\boldsymbol{P}_{\text{cs}}$ is visualized as a vector field. Since the price vector is just a normalized gradient of the trading function φ_{cs}, $\boldsymbol{P}_{\text{cs}}$, it is necessarily orthogonal to the invariant curves. All of this can be seen in Figure 2.

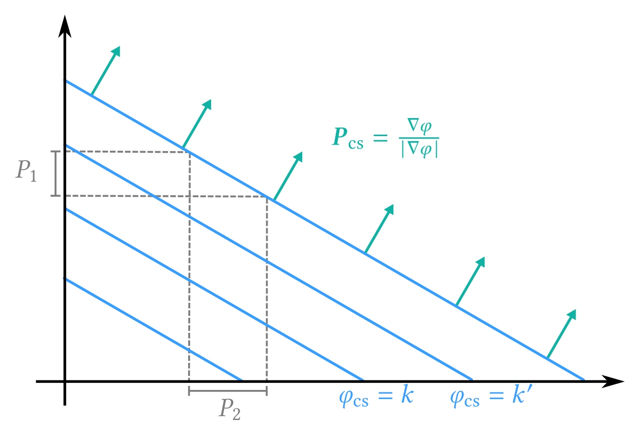


Figure 2. Limit Order as a Constant Sum Market.

We can now look at the invariant curves and the constant price curves for Uniswap V2 and RMM-01. The invariant curves consist of the attainable prices for a swapper assuming no fees. By trading a swap with no fees, the values of the reserves are required to change so that the invariant remains constant in Uniswap V2 and so that the number of LPTs remain constant for RMM-01. The constant price curves are the curves that an LP will follow. The amount of reserves tendered or received must maintain the price of the CFMM, hence the name of constant price curves. Moreover, valid liquidity changes will only change the invariant in Uniswap

V2 and the amount of (or value of) LPTs in RMM-01. Please see the curves in Figure 3.

When there is an included fee γ, a swapper will swap $\gamma\Delta$ and provide $(1-\gamma)\Delta$ as a liquidity change. That is, $(1-\gamma)\Delta$ will be added by following the constant price curves while $\gamma\Delta$ will be swapped along the invariant curves. This fee will necessarily increase the invariant in Uniswap V2, while in RMM-01 the fee will increase the amount of reserves that an LP receives when they cash-out their LPT.

B Price Impact and Comparison

The price impact of trades on RMM-01 depend heavily on the choice of σ and τ. For example, we will always see that as $\tau \to 0$, the price impact approaches zero as well. We can see this algebraically, but visually we have

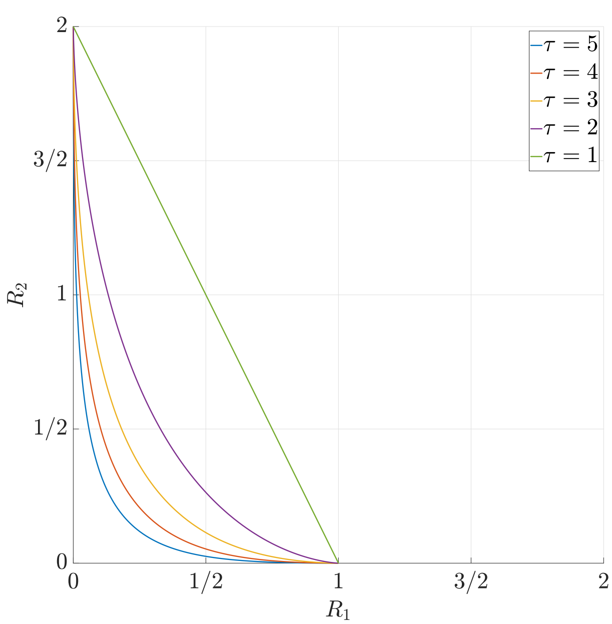


Figure 4. The change in the RMM-01 trading curve as τ varies. We have otherwise fixed $K = 2$ and $\sigma = 1$.

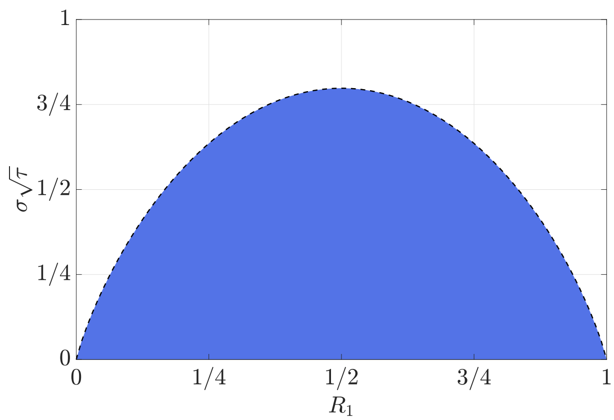


Figure 5. The values for $\sigma\sqrt{\tau}$ where RMM-01's price impact is less than Uniswap V2.

Glossary

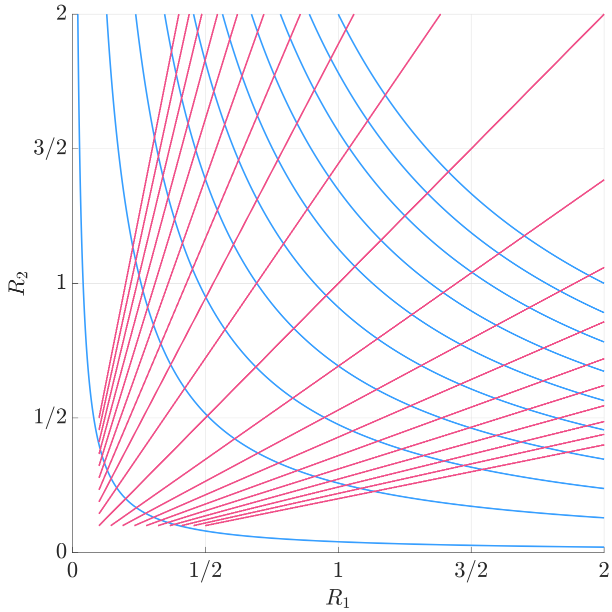
ERC20 The standard used for creating smart contracts on the Ethereum blockchain.

RMM-01 Primitive's implementation of a CFMM that replicates the payoff of a covered call.

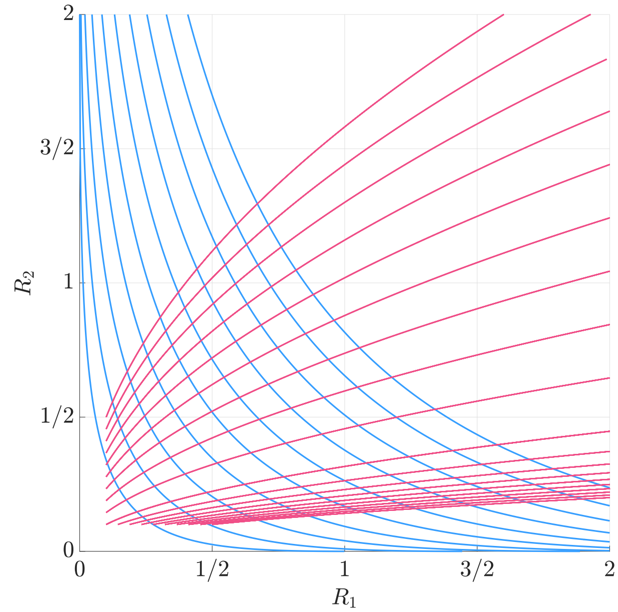
Uniswap V2 A constant product market maker based exchange. Note that there has been an update to their protocol to Uniswap V3 which is described in their white paper [2].

Acronyms

LPT Liquidity Provider Token
CC Covered Call
CFMM Constant Function Market Maker
DeFi Decentralized Finance
DEX Decentralized Exchange
LP Liquidity Provider
RMM Replicating Market Maker
TradFi Traditional Finance



(a) Curves for Uniswap V2.



(b) Curves for RMM-01 with $K = 2$, $\sigma = 1$, and $\tau = 5$.

Figure 3. The invariant and constant price curves for two CFMMs.