Poption

A General-Purpose Exotic Option Designed For DeFi

hydrogenbear*

July 20, 2022

Abstract

In this paper, a new financial derivative, Poption, is proposed. It is a general-purpose exotic option combined with ideas from the prediction market. It would be very convenient to construct European options, binary options, margin trading, and other more complex derivatives from Poption. The invention has the advantages of simple structure and powerful functions. For financial markets, capital-efficient derivatives are essential components. Poption can be the basic structure of future blockchain derivatives. In this paper, we will introduce the Poption contract, constant function market maker on Poption, pricing Poption based on the prediction market theory and Black-Scholes model, and how to build derivatives markets based on Poption.

Contents

*hydrogenbear@poption.org

1 Introduction

DeFi (distributed finance) has achieved great success in the spot market over the past two years. Distributed exchanges such as Uniswap[2] and Sushiswap already have a large market. However, due to the lack of a trading framework that adapts to blockchain and smart contracts in derivatives trading. The derivatives market has not flourished in DeFi like the spot market. Centralized exchanges such as Binance still dominate the derivatives market, and dYdX[10] is not really a distributed exchange. Poption is a set of derivatives solutions designed for DeFi and smart contracts. From the bottom up, it is based on mathematics and computer science, in line with software engineering, highly flexible and robust. From a high-level perspective, it can be used to build European options and binary options, and simulate leveraged trading and other financial products. It gives us the ability to decouple the development of complex financial products and complex blockchain applications and achieve these goals.

1. To enable a variety of different financial derivatives to share the same liquidity pool. This will improve liquidity and reduce slippage.

2. To make it simple for financial teams to use smart contracts to price risk and make money on DeFi.

This article is divided into the following sections for detailed discussion.

1. Introduce Poption, discuss its properties, and discuss how it can be used to form or simulate other financial derivatives such as European options.

2. Introduce a Poption market-maker based on constant function market maker. Discuss the loss faced by market makers and the countermeasures that need to be taken.[4]

3. Discuss how to price Poption from the perspective of prediction market theory and the Black-Scholes model.[8, 12]

4. Discuss how to establish the market and the respective roles and motivations of market participants.

1.1 Previous Work

There are already many options products on the blockchain. Opyn[11], Ribbon[9], and Lyra[6] are the most successful products in the current Ethereum-compatible chain. Opyn and Ribbon together form a system, while Lyra is a system on its own. Their success validated derivatives market demand on DeFi. Among them, Ribbon provides earning pools and their trades heavily rely on OTC. Lyra is an auto market maker but not based on ERC20 tokens, which is a strong restriction. In addition, they are all based on collateral, which put participants at risk of forced liquidation. Their design is based heavily on the existing financial products, which makes the product too complicated in DeFi, and cannot form a large ecosystem. Our solution will build an options retail market based on ERC20 tokens simply and elegantly. This will complement the market ecosystem.

2 Poption Contract

The name poption is derived from 'payoff option', 'prediction option', and 'popular option'. As the name suggests, it is a kind of option focusing on payoff functions. Poption contract is a derivative that allows investors to customize the payoff in a very simple framework. This enables Poption to be flexible enough to meet different financial needs in the same market. Poption contracts will always be covered. There is no liquidation risk in it. It will be simple to price and trade Poption on blockchains. We will introduce Poption contracts and Poption smart contract in this section. We will also introduce how to use Poption to simulate some of the common financial derivatives like binary options, European options, etc.

2.1 Definition

A Poption contract is defined by an settlement asset (or currency) A , underlying oracle (given the price of underlying asset) at time t $S(t)$, a payoff function $f(x)$ and an expiration moment in the future T. It stipulates that a Poption contract holder will obtain $f(S(T))$ shares of asset A, after a future moment T. The contract can be denoted as $Poption(A, S, T, f)$. And the definition can be written as $t_{current} \geq T \Rightarrow Poption(A, S, T, f) = f(S(T))A$. T is the expiration moment. $S(T)$ is the settlement price. $t_{current}$ is the current time. The settlement price will not change once it is determined after the expiration moment. Poption can be regarded as a prediction market contract whose event is the price at a future time T. It has the following characteristics.

1. Only transact with the settlement asset. In Poption, except for the price information $S(T)$, the underlying asset is not involved in a transaction,

and there will be no holding, buying, or selling of the underlying asset. This feature gives Poption simplicity.

2. By defining different $f(x)$, we can meet different financial needs under the same framework. Poptions with different payoff function can be traded by the same constant function market maker. This makes Poption powerful.

3. Compared with other financial products, Poption has a 'closed form' and simple structure. The payoff of the contract is only determined by the payoff function and the settlement price. Unlike derivatives that have liquidation mechanism, the payoff of Poption is unrelated to the price before the expiration moment.

Unless otherwise stated, A, C, S, T represent the underlying asset, cash, underlying asset price denominated in cash, and expiration moment, respectively, below in this article.

2.2 Mint and Add rule

The minting, spliting and merging of Poptions shall follow these two rules.

- **Mint Rule** We can mint a contract $Poption(A, S, T, f)$ from any c shares of settlement asset A, where $c \geq 0$ and $f(x) := c$ is a constant function. This rule can be denoted as $c \geq 0 \Rightarrow Poption(A, S, T, c) = cA$. We also use the rule when we want to burn some Poptions.

- **Add Rule** We can split a contract $Poption(A, S, T, f)$ to $Poption(A, S, T, g)$ and $Poption(A, S, T, h)$, where $f = g + h, g \geq 0, h \geq 0$. This rule can be denoted as $f = g + h, g \geq 0, h \geq 0 \Rightarrow Poption(A, S, T, f) = Poption(A, S, T, g) + Poption(A, S, T, h)$. We also use the rule when we want to merge some Poptions.

2.3 Poption Smart Contract

In order to implement Poption on blockchain, we introduce Poption smart contract. It will provide the methods of minting, burning, transferring and exercising. We will also prove that all the poption in such a smart contract can always be exercised when they expire.

A Poption smart contract has three basic properties, settlement asset A , expiration moment T, and price oracle $S(t)$. All Poptions in the same smart contract share these three properties. This makes it possible to split and merge Poptions in the contract. Besides these properties a Poption smart contract consists of an asset pool aA, a group of users $i \in U$ and their corresponding payoff function f_i. The smart contract should support the following method.
At initialization:

- `init(A, T, S)` : It records the settlement asset A, the expiration moment T, and the (oracle) method of obtaining the price S. It initializes all payoff functions $f_i = 0$. There is $0A$ in the asset pool.

Before expiration:

- `mint(sender, c)` : It transfers the sender's asset cA into the smart contract, and the sender's payoff function becomes $f_{sender}(x) := f_{sender}(x) + c$. This method is based on mint rule.

- `burn(sender, c)` : If sender's payoff function $f(x) \geq c$, then sender's payoff function becomes $f_{sender}(x) := f_{sender}(x) - c$, and transfers the assets cA from the smart contract to the sender. This method is also based on mint rule.

- `transfer(sender, receiver, g)` : If the payoff function of the sender $f_{sender}(x) \geq g(x)$, then the payoff function of the sender subtracts g, which is $f_{sender}(x) := f_{sender}(x) - g(x)$, and the payoff function of receiver adds g, which is $f_{receiver}(x) := f_{receiver}(x) + g(x)$. This method is based on add rule.

After expiration:

- `exercise(sender)` : It transfers $f_{sender}(S(T))$ shares of assets A from the smart contract to the sender, and sets $f_{sender}(S(T)) := 0$.

Next we will prove that all Poption contracts in the smart contract are always exercisable.

- In the initial state, $a = 0, \forall i \in U, f_i = 0$, so $\sum_{i \in U} f_i = a$.

- If $\forall i \in U, f_i \geq 0$ and $\sum_{i \in U} f_i = a$, it is apparent to see that after executing `mint`, `burn` or `transfer`, $\forall i \in U, f_i \geq 0$ and $\sum_{i \in U} f_i = a$ still holds.

- If $\forall i \in U, f_i \geq 0$ and $\sum_{i \in U} f_i = a$ are true before expiration moment, then $\forall i \in U, f_i \geq 0$ and $\sum_{i \in U} f_i(S(T)) = a$ hold after expiration moment.

- If $\forall i \in U, f_i \geq 0$ and $\sum_{i \in U} f_i(S(T)) = a$, then after executing the `exercise` method, $\forall i \in U, f_i \geq 0$ and $\sum_{i \in U} f_i(S(T)) = a$ still hold.

- If $\forall i \in U, f_i \geq 0$ and $\sum_{i \in U} f_i(S(T)) = a$, then $\forall i \in U, f_i(S(T)) \leq a$ which means all contracts in the smart contract are exercisable.

Therefore, all contracts in the Poption smart contract are exercisable. This means both buyers and market makers of Poption will never be liquidated and can ignore the squeeze risk or death spiral that can occur in other derivative. This will simplify the trading and pricing, which is of great help when we want to create an automatic market maker.

2.4 Correspondence Between Poption and Other Financial Derivatives

Unlike other derivatives that usually achieve high capital efficiency through a margin account, we gain high capital efficiency and flexibility by defining different payoff functions in Poption. In this section, we will introduce the relationship between Poption and some other financial derivatives. Prove that we can turn complex financial operations into operations of mathematical functions. Financial operations require many transactions. Meanwhile, mathematical operations cost almost nothing.

2.4.1 Binary Options

By defining the payoff function as a shifted unit step function we can get binary options. First, we define the binary payoff function $binary_K(x) := \begin{cases} 0 & x < K \\ 1 & x \geq K \end{cases}$. Then we can represent all four types of binary options as Poptions.

- $Poption(A, S, T, binary_K)$ is an asset-or-nothing call binary option with strike price K.
- $Poption(A, S, T, 1 - binary_K)$ is an asset-or-nothing put binary option with strike price K.
- $Poption(C, S, T, binary_K)$ is a cash-or-nothing call binary option with strike price K.
- $Poption(C, S, T, 1 - binary_K)$ is a cash-or-nothing call binary option with strike price K.

2.4.2 European Options

Poptions can simulate European options. There are two simulation methods. One is to use binary options. For example, a European call option can be obtained by buying asset-or-nothing call and cash-or-nothing put.[8] Another approach, which we will discuss in detail, is to use payoff function f directly to simulate the target derivatives.

1. **Put Options** The payoff function of a European put option with strike price K can be expressed as $put_K(S) = max(0, K - S)$. So $Poption(C, S, T, put_K)$ is a Poption contract which simulates the European put option.

2. **Call Options** The payoff function of a European call option with strike price K can be expressed as $g(S) = max(0, S - K)$. Then $Poption(C, S, T, g)$ is a call option with strike price K. Here we have a problem. g is a function with no upper bound, so minting a Poption that can split this call option requires infinite cash, which is impossible.

   We can solve this problem by changing the settlement asset of Poption to the underlying asset A. The payoff function with the unit of a call option can be expressed as $max(0, S(T) - K)C = max(0, S(T) - K)\frac{A}{S(T)} = max(0, 1 - \frac{K}{S(T)})A$, Let $call_K(S) = max(0, 1 - \frac{K}{S})$, then $Poption(A, S, T, call_K)$ is a covered European call option with strike price K. Here $call_K$ is a bounded function, which means we can get such a Poption by minting and spliting.

In finance, when we write naked options, similar problems occur. It is called gamma squeeze.[3, 13]The design of Poption excludes squeeze and invalidates such options. However, naked options are still very common in the market. For such requirements, we can define $Poption(C, S, T, f)$, where $f(x) = min(max(0, x - K_0), K_1), K_0 < K_1$ to meet the needs without introducing squeeze. This actually equals to the bull spread strategy in option.

2.4.3 Margin Trading

We cannot synthesize margin trading perfectly in Poption. However, we can still simulate the value of the margin account at expiration moment T. Suppose the leverage is k, the current price is $S(0)$, and we buy 1 unit of the asset on margin. If the margin account has never been liquidated before the expiration moment, or the asset price remains unchanged after the margin account is liquidated. Then at the expiration moment T , the value of the user's margin account should be the value of the asset minus the value of the borrowed cash, which is $max(0, S(T) - \frac{k-1}{k} S(0))$, that is, at this time, the account value equals the value of a call option with a strike price of $\frac{k-1}{k} S(0)$. Therefore, we can use Poption to synthesize a call option to simulate this margin trading. Aside from margin calls, the main difference between these two approaches is cost. The cost of obtaining such a margin account is $\frac{S(0)}{k}$ plus some interest and other fees. The cost of a call option is determined by the option seller. If the price of the corresponding Poption is similar or lower, the Poption would be a more competitive product.

2.4.4 Impermanent Loss Hedging

Suppose a market maker in Uniswap[1] has r_c cash and r_a asset in the liquidity pool and the current asset price is $S(0) = \frac{r_c}{r_a}$. The impermanent loss of the market maker is $r_c + S(T)r_a - 2\sqrt{S(T)r_c r_a}$ when the price goes to $S(T)$. let $f(x) = max(r_c + r_a x - 2\sqrt{r_c r_a x}, r_c)$, $f(S(T))$ is equal to the impermanent loss when $0 \le S(T) \le 4S(0)$, so we can use $Poption(C, S, T, f)$ to hedge the impermanent loss. If we needs to hedge against the risk of a higher price increase, we can add $Poption(A, S, T, g)$ into the portfolio to hedge the rest loss, where $g(x) = \begin{cases} 0 & x \le 4S(0) \\ r_a - 2\sqrt{\frac{r_c r_a}{x}} & x > 4S(0) \end{cases}$.

2.4.5 Off-Chain Assets

Like naked options, we can build derivatives of off-chain assets through Poption without holding the off-chain assets. We can use oracles to obtain the price of off-chain assets and deploy Poption smart contracts to help users access a specific market which is not accessible before. For example, we can construct $Poption(\text{USDC}, S, T, f)$ based on USDC, a future moment T and the oracle S of Apple's stock price. If $f(x) = min(max(0, x - K_0), K_1)$, the Poption equals a bull spread strategy of Apple, it will satisfy the need to long the stock. On the other hand, we can also use $min(max(0, -x + K_1), K_0)$ to construct a bear spread strategy to satisfy shorting needs.

2.4.6 Short

In traditional financial markets, to short options are more complicated than to long options. It requires a collateral account to ensure that when the option owner exercises the option, the writer should hold or can buy enough assets from the market to pay the exerciser. In the Poption contract, this is simplified. Holding c assets in Poption and selling a Poption contract with a payoff function of $f(x)$ is equivalent to buying a Poption contract with a payoff function of $c - f(x)$. In other words, any forms of shorting can be done by directly longing the Poption contract of a specific function.

2.4.7 Synthesize

In traditional financial markets, it is common to use basic financial derivatives to synthesize more complex derivatives to meet financial needs. For example, in the traditional market, a butterfly option[8] can be formed by buying a call option with a strike price of K_0, selling two call options with a strike price of K_1, and then buying a call option with a strike price of K_2, where $K_0 + K_2 = 2K_1$. In practice it is complicated. In Poption, investor only needs to buy Poption whose payoff function is $call_{k_0} - 2call_{k_1} + call_{k_2}$ to get the butterfly option, which is very convenient.

Sometimes when we synthesize a payoff function that $inf(f(x)) \neq 0$, we can use $f'(x) := f(x) - inf(f(x))$ to get a payoff function with a maximum lower bound of 0. The new payoff function can form a valid and most capital efficient Poption contract with equal net profit to the previous contract. In traditional finance, this is equivalent to mortgaging the income that can always be obtained from the synthesized derivatives to get spot assets or cash. This is more complicated in practice, but it can be done very simply in Poption.

2.5 Discretization and Payoff Tokens

In implementation, the continuous payoff function f is not friendly. Therefore, we need to discretize f so that we can implement a Poption smart contract on the chain. We can achieve this by approximating the payoff function with a piecewise linear curve. Specifically, we have N strike points $s_1, s_2, s_3, \ldots, s_N$, where $\forall i > 0, s_i > s_{i-1}$. We can use a piecewise linear curve

$$f'(x) = \begin{cases} f(s_1) & x < s_1 \\ f(s_i) & x = s_i \\ f(s_N) & x \geq s_N \\ \frac{s_{i+1} - x}{s_{i+1} - s_i} f(s_i) + \frac{x - s_i}{s_{i+1} - s_i} f(s_{i+1}) & s_i \leq x < s_{i+1} \end{cases} \tag{1}$$

to approximate the original payoff function f . Then the Poption on the original continuous domain is discretized. We can use the vector $\boldsymbol{a} = [f(s_1), f(s_2), \ldots, f(s_N)]$ to represent the Poption with the piecewise linear curve held by the holder and greatly simplify the calculation. In practice, the vector $\boldsymbol{a}$ is implemented as tokens of an ERC1155 contract. We call them **payoff tokens**. The element a_i of $\boldsymbol{a}$ is stored in the ERC1155 contract as token balance and can be read via `balanceOf(owner, i)` method.

Theoretically, for any bounded function f, we can always find a piecewise linear curve f' with $n+1$ segments to approximate it such that $|f'(x) - f(x)| \leq \frac{1}{2n}(sup(f(x)) - inf(f(x)))$. For non-monotonic f, we can divide it into several monotonic intervals for discussion, and the conclusion does not change in a single monotonic interval.

In fact, how to set the strike points, what kind of error and computational complexity are acceptable, need to be determined by market demand. If the market demand for Poption accuracy cannot be met, we need to set more strike points, and if the demand for computational complexity cannot be met, we should set fewer strike points.

In addition, the discretized Poption still retains an accurate representation of a spread strategy with strike prices $s_1, \ldots, s_{n-1}$.

3 Automatic Market Maker

After we mint the Poption contract, we want to be able to trade it. Just splitting an asset into two derivatives does not make sense to the holder. Since Poption itself can be viewed as a prediction market contract, the foundational research of many automatic market makers comes from prediction markets.[7] Naturally, we can use these market maker mechanisms to make the Poption market. Here we will show how to build a constant function market maker with Poption. The general idea is to treat each payoff token as an independent asset, and then apply a multi-asset constant function market maker on them.

3.1 Constant Function Market Maker

Constant function market makers are the most common and successful type of automated market makers in DeFi. They rely on a trade function to decide whether or not a trade should take place. We will show how to use weighted geometric mean to construct a constant function market maker for trading Poptions with the same settlement asset A, the target price S and expiration moment T . Since all Poption contracts in a transaction share the same A, S, T, the notation of the payoff tokens $\boldsymbol{a}$ also represents the Poption $Poption(A, S, T, \boldsymbol{a})$ in this section.

Define the trade function as $\phi(\boldsymbol{w}, \boldsymbol{a}) = \prod_{i=1}^{n} a_i^{w_i}$, where $\boldsymbol{w}$ is the weight vector, and $\boldsymbol{a}$ is the balance of payoff tokens.[4] For a transaction, assume that the Poption in the market maker's liquidity pool is $\boldsymbol{r}$, the Poption the investor wants to buy is $\boldsymbol{\delta}$, and the Poption the investor wants to sell is $\boldsymbol{\epsilon}$. If $\phi(\boldsymbol{w}, \boldsymbol{r} - \boldsymbol{\delta} + \frac{\boldsymbol{\epsilon}}{1+\gamma}) \geq \phi(\boldsymbol{w}, \boldsymbol{r})$, the market maker trades with the investor, otherwise the transaction is rejected, where γ is the fee rate charged by the market maker. The liquidity pool after the transaction becomes $\boldsymbol{r} - \boldsymbol{\delta} + \boldsymbol{\epsilon}$. In this way we have constructed an automatic market maker based on the weighted geometric mean.

3.2 Buy and Sell Poption

In Poption, investors may wish to buy or sell Poption directly. For such buyers, the above transaction condition becomes $\phi(\boldsymbol{w}, \boldsymbol{r} - \boldsymbol{\delta} + \frac{c}{1+\gamma}) \geq \phi(\boldsymbol{w}, \boldsymbol{r})$, where c is the cost of purchasing a Poption in settlement asset. This transaction equals to mint cA to $Poption(A, S, T, c)$ and swap it with market maker for $Poption(A, S, T, \boldsymbol{\delta})$. For buyers looking to buy $\boldsymbol{\delta}$, $\boldsymbol{w}, \boldsymbol{r}, \boldsymbol{\delta}$ are all known. By solving the equation $f(x) = \phi(\boldsymbol{w}, \boldsymbol{r} - \boldsymbol{\delta} + \frac{x}{1+\gamma}) - \phi(\boldsymbol{w}, \boldsymbol{r}) = 0$ we can get the cost $c = x$ to buy the Poption. This equation can be easily solved by Newton's method.[5] Similarly, we can get the return of selling the Poption by solving the trading conditions equation $\phi(\boldsymbol{w}, \boldsymbol{r} - x + \frac{\boldsymbol{\epsilon}}{1+\gamma}) = \phi(\boldsymbol{w}, \boldsymbol{r})$.

3.3 Price

In constant function market makers of Poption, price of each payoff token is defined as the ratio of the cost to the payoff token in a minimal transaction. So for the payoff token i, its price is $\frac{\partial - c}{\partial \delta_i}$ when $\boldsymbol{\delta} \to \boldsymbol{0}, c \to 0$.

When buying a Poption, the price is

$$\frac{\partial - c}{\partial \delta_i} = \frac{\dfrac{\partial \phi(\boldsymbol{w}, \boldsymbol{r} - \boldsymbol{\delta} + \frac{c}{1+\gamma})}{\partial \delta_i}}{\dfrac{\partial \phi(\boldsymbol{w}, \boldsymbol{r} - \boldsymbol{\delta} + \frac{c}{1+\gamma})}{\partial - c}} = (1+\gamma) \frac{\frac{w_i}{r_i}}{\sum_{i=1}^{n} \frac{w_i}{r_i}} \tag{2}$$

.

When selling a Poption, the price is

$$\frac{\partial - c}{\partial \epsilon_i} = \frac{\dfrac{\partial \phi(\boldsymbol{w}, \boldsymbol{r} - c + \dfrac{\boldsymbol{\epsilon}}{1+\gamma})}{\partial \epsilon_i}}{\dfrac{\partial \phi(\boldsymbol{w}, \boldsymbol{r} - c + \dfrac{\boldsymbol{\epsilon}}{1+\gamma})}{\partial - c}} = \frac{\dfrac{w_i}{r_i}}{(1+\gamma)\sum_{i=1}^{n} \dfrac{w_i}{r_i}} \tag{3}$$

.

We define the mid-price of the payoff token i as $\boldsymbol{p} = [p_1, p_2, \ldots, p_n]$ where $p_i = \dfrac{\frac{w_i}{r_i}}{\sum_{i=1}^{n} \frac{w_i}{r_i}}$. It can be easily proved that $\sum_{i=1}^{n} p_i = 1$. That is, the sum of the mid-price of all interval i contract is 1. This is consistent with the mint rule.

3.3.1 Adjust Price

The mid-price of Poption needs to be adjusted in time as time goes by and the spot price changes. In the Poption constant function market maker, we adjust the price by changing the weight $\boldsymbol{w}$. For the current liquidity pool $\boldsymbol{r}$, if we expect to adjust the price to $\boldsymbol{p}$, then we should set the weights to $w_i = \dfrac{p_i r_i}{\sum_{i=1}^{n} p_i r_i}$.

3.4 Arbitrage and Loss

When there is a difference between the price given by the market maker and the real value of the Poption, there is opportunity for arbitrage. These arbitrage usually result in loss of the market makers. This is impermanent loss in spot market but it will be permanent when Poptions expire in Poption market. To study the loss, we can describe the arbitrage by solving an optimization problem. Suppose the real value of each interval of the Poption is $\boldsymbol{q} = [q_1, q_2, \ldots, q_3]$, we can find a transaction $\boldsymbol{\delta}$ in the tradable space and maximize its value. The specific optimization problem is as follows.

$$\begin{aligned} \max_{\boldsymbol{\delta}} \quad & \sum_{i=1}^{n} q_i \delta_i \\ \text{s.t.} \quad & \phi(\boldsymbol{w}, \boldsymbol{r} - \boldsymbol{\delta}) \geq \phi(\boldsymbol{w}, \boldsymbol{r}) \end{aligned} \tag{4}$$

The first line of the problem represents maximizing the value of the transaction, which is also the loss caused by arbitrage to the market maker. The second line of the problem limits the transaction within the tradable space. Here we ignore the fee.

Solving this problem by Lagrange multiplier[5], we can get the analytical solution as follows, $\delta_i = r_i - \alpha \dfrac{w_i}{q_i}$, where $\alpha = \prod_{i=1}^{n} (\dfrac{q_i r_i}{w_i})^{w_i}$. Then the arbitrage value is $\sum_{i=1}^{n} q_i \delta_i = \sum_{i=1}^{n} r_i q_i - \alpha w_i$. To see it more clearly, we substitute $\frac{w_i}{r_i}$ with the middle price p_i via the relation $\frac{w_i}{r_i} = p_i \sum_{j=0}^{n} \frac{w_j}{r_j}$. Then the value becomes $\sum_{i=1}^{n} (q_i - p_i \prod_{j=1}^{n} (\frac{q_j}{p_j})^{w_j}) r_i$. This is a general result, and we will discuss this result in two special cases.

1. When $\boldsymbol{p} = \boldsymbol{q}$, the arbitrage value is 0. This is easy to understand, if there is no spread between the true value and the price then there is no room for arbitrage.
2. When one item in $\boldsymbol{q}$ approaches 1 and the other items approach 0, that is, $q_j \to 1, \forall i \neq j, q_i \to 0$, if $\forall i, w_i > 0$, then the arbitrage value is close to r_j. Since $q_j \to 1$, total value in the liquidity pool is also close to r_j . In other words, the market maker may lose all of his value in the liquidity pool. Since the settlement asset required to execute such an arbitrage is approaching infinity, this situation would not happen. But it is still realistic if the arbitrage target is only most of the value in the liquidity pool.

The second situation occurs in Poptions. Because when $t \to T$, one of $\boldsymbol{q}$ will approach 1. Accurate pricing and appropriate fee can help us avoid this problem. To execute such a transaction $\boldsymbol{\delta}$ we need to buy $\boldsymbol{\delta} - min(\boldsymbol{\delta})$ and pay $-(1 + \gamma)min(\boldsymbol{\delta})$. Then the fee required for the transaction is $-\gamma min(\boldsymbol{\delta})$. Arbitrage occurs only when the arbitrage value is greater than the fee. We can estimate the maximum loss when there is a fee. For the rate γ, even if the arbitrageur can completely determine the price $S(T)$ at the future expiry time, he will generally get at most the value of $(1 - (1 + \gamma)p_j)r_j$. In addition, we can also close the market early to reduce the occurrence of q_j approaching 1. One caveat is that even high fee rates cannot make up for these loss when the pricing is poor. For example, in the above formula, when $\gamma = 1, p_j = 0.4$, the maximum loss is 0.2, that is, the loss still exists when the rate is as high as 100% but the pricing error is 60%. Therefore, we need more precise pricing.

4 Pricing

Due to the existence of arbitrage loss, pricing is very important for a Poption market maker. In this section we will describe how to price Poptions. We will introduce this part from two aspects. On one hand, it starts from the prediction market, and on the other hand, it starts from the Black-Scholes model commonly used in option market. In the end we will come to the same conclusion.

4.1 Prediction Market Pricing

The theory of predicting market pricing is based on information aggregation. In prediction markets, the value of a predicted event contract is equal to the probability of that event happening. Under the effective market maker mechanism, the price of the contract will tend to the probability of the event as trading keeps happening. Market makers are buying information about this probability from market participants.[7] Conversely, if the initial contract price is equal to the probability of the event, the market maker will have no loss from buying information. If we denote the probability of an event as Pr_i , then the contract value of the predicted corresponding event should also be Pr_i . In Poption, we need to use risk-neutral probabilities instead of physical probabilities to price contract due to the financial properties of Poptions.

To understand this, we can study a simple example. Suppose the current price of asset A is $S(0)$, denominated in cash C. At the future moment T, the probability of the price rising relative to the current price is 50%, and the probability of falling is also 50%, namely $Pr(S(T) \geq S(0)) = 0.5, Pr(S(T) < S(0)) = 0.5$. A market maker predicts this fact very accurately. If he is pricing $Poption(C, S, T, f)$ and $Poption(A, S, T, f)$ according to the physical probability, then the arbitrageur can use $S(0)C$ to buy $2S(0)$ cash-or-nothing put options $(Poption(C, S, T, 2S(0)(1 - binary_{S(0)})))$; and buy 2 asset-or-nothing call options $(Poption(A, S, T, 2binary_{S(0)}))$ with $1A$. At expiration, if the price falls, he will receive $2S(0)$ in cash. If the price goes up then he will get 2 shares of the settlement asset whose value is more than $2S(0)$. In this way, arbitrageurs can always profit regardless of whether the price rises or falls. To avoid this happening, we must use risk-neutral probabilities to adjust the risk. After using risk-neutral probability, the pricing of Poption is the same as that in an ordinary prediction market. If the risk-neutral probability density of the settlement price is $Q(S(T))$, then the middle price corresponding to the Poption with payoff f should be $p = \int_0^\infty f(x)Q(x)dx$.

Unlike normal prediction market, poption auto market maker can use the information from the spot market to avoid arbitrage loss. We utilize these information via Black-Scholes model.

4.2 Black-Scholes Model

The Black-Scholes (BS) model is a basic model for option pricing in finance, and it is also the most widely used pricing model. We can use this to price Poptions. In the BS model, the pricing formulas of binary options are known. From them, we can deduce the pricing formula of Poption.

4.2.1 Pricing of Poption With Cash as Settlement Assets

The value of one unit of cash-or-nothing call binary option with strike price K is $DPr(S(T) \geq K)$,[8, 12]where D is the discount rate, here we simply do not consider the discount and set it to 1. $Pr(S(T) \geq K)$ is the exercise probability calculated under the assumption that future asset prices follow a log-normal distribution and the expected value of future asset price is the current asset price. Generally, it is denoted as $N(d_2)$ in the BS model, and here we are more concerned about its meaning of probability, so it is denoted as $Pr(S(T) \geq K)$. This is consistent with the conclusions in the information market.

Regarding discounting, if we design the settlement assets as discounted assets, such as bonds that can provide market risk-free interest rates, we can naturally get discounts without changing the pricing method. This means discounting can be decoupled from Poption by wrapping tokens.

4.2.2 Pricing of Poption With Asset as Settlement Assets

The value of one unit of asset-or-nothing call binary option with strike price K is $S(0)Q(S(T) \geq K)$. $Q(S(T) \geq K)$ is the risk-neutral exercise probability calculated under the assumption that future asset prices follow a log-normal distribution and the expected value of future asset price is the current asset price. Generally, it is denoted as $N(d_1)$ in the BS model, and here we are more concerned about its meaning of risk-neutral probability,[12] so it is denoted as $Q(S(T) \geq K)$. This is also consistent with the conclusions in the information market.

4.3 On-chain Pricing and Off-chain Pricing

The BS model is simple enough that pricing can be done on-chain.[6] This pricing is real-time pricing and can be executed in the transaction. This is also the solution of some existing DeFi option products. However asset prices do not always obey the assumptions of the BS model. At this time, the price given by the BS model will deviate. The information market theory tells us that the accuracy of pricing is very important. Only with accurate pricing can we reduce loss of liquidity providers and achieve lower fee rates and higher transaction volumes. If we can use off-chain information and computing resources, it would be helpful.

In the previous section, we discussed how to calculate possible loss when the price given by the market maker does not match the true value. When the fee rate set by the market maker cannot cover this loss, there is a risk of arbitrage. Conversely, when the rate is sufficient to cover the loss, there is no arbitrage risk even if the price is not adjusted in real time. Therefore, there is an off-chain pricing strategy, that is, when the change in the value of Poption is not enough to cause a loss higher than the fee or threshold, we can choose not to change the parameters of the BS model and the CFMM. When the value of Poption changes greatly, which will bring risk or affect profit, the market maker shall update the parameters of the BS model and the CFMM. This mechanism will enable the market to use a complex pricing system off-chain and update the parameters on-chain when needed. This relieves the computational pressure of online valuations and protects the intellectual property of the market maker. Of course, it also has drawbacks, but they can be overcome.

- For a market maker using public liquidity pool, due to the lack of financial supervision on the chain, if parameters can be tuned, the market maker pricing team can set prices maliciously and withdraw assets from users, similar to front-running. This problem can be solved in two ways. 1. Add a proof method to the contract on the chain, and the contract refuses to accept the pricing when the pricing does not pass the proof method. This proof can be a complete verification of the offline calculation process or it can be looser and only play a role in proving the harmlessness of pricing. 2. it can also be solved through game theory. We can add a delay to the effectiveness of the new parameters. If other market makers in the market are also ready to trade according to their own pricing method, even if there will be unreasonable pricing, it will be quickly detected before effectiveness. The pricing team would not be able to front-run.

- If the centralized off-chain pricing system crashes or disconnects or the market changes too quickly, the new pricing parameters will not be updated in time, which may result in loss of liquidity pools. To solve this problem, some mechanism needs to be added to the on-chain contract. For example, we can introduce circuit breaker to solve such problems.

5 Create Market

We already have the necessary tools to create markets. In our vision, this would be an ecologically distributed market, with multiple market makers and multiple front-end platforms serving multiple kinds of customers. Next, we will discuss the roles and motivations of the various players in the market, and analyze their needs and the value the market can bring to them.

5.1 Market Maker

Market makers are the backbones of the market, and they are primarily responsible for providing liquidity and pricing. A market maker can be an on-chain contract that relies entirely on a public liquidity pool. It can also be a fund team that operates offline with its own funds. They can hedge or not hedge their positions. They are market makers as long as they can provide pricing and liquidity to the market. In addition, the first market maker in the market is also responsible for determining the A of the settlement asset of the contract, the price S , the expiration moment T ,the intervals $[s_{i-1}, s_i)$ and create the contract pool. Market makers' profits come from the fee charged in each transaction, and market makers' expected loss come from the cost of buying information from market participants. Market makers need to compete with each other, and market makers that can provide higher liquidity, lower fees and guarantee their own returns will survive. We need to build a good infrastructure so that market makers can make markets without much contract or software development knowledge.

5.2 Hedger

Hedging risk is a common need in the derivatives market. For example, Market makers in the spot market may use Poption to hedge their impermanent loss.

Hedgers don't care about future changes in asset prices, they want to lock in the total portfolio loss when asset price changes. In other terms, when a hedger buys Poption, he is buying insurance from a market maker. At this time, market makers play an actuarial role, providing insurance products to hedgers by accurately calculating the probability of the price falling into a certain interval. The insurance is guaranteed that the hedger's loss will always be the same regardless of future price changes. If the market maker's pricing is accurate, the fee cost will be added to the hedger's expected loss. But hedgers are shielded from the risk of asset price volatility. For them, we need to provide a powerful front-end and good marketing, so that they can find products that meet their needs in Poption.

5.3 Speculators

In markets, in addition to setting prices by gathering information directly from the outside world, market makers also buy information from market participants. These market participants are called speculators. They gather information from everywhere and submit it to the market, making market prices more accurate. Speculators who can provide market makers with more right information can make profits from the market, otherwise they will not only lose transaction fees, but also pay for providing wrong information.

Generally speaking, arbitrageurs in the market play this role, and a market maker can also become a speculator of other market makers. For some excellent speculators, they will continue to win value from the market and form a more stable methodology for pricing, at some point they may transform into market makers. Such a transform path should be provided. In addition, speculators provide information and hedgers consume information, so their requirements for market makers are different, and market makers need to balance their own market-making strategies to meet the different needs from both.

6 Current Work and Future Work

At present, we have completed the development of Poption contract pool, automatic market maker, and pricing contract in the Polygon. We also developed a web front-end with functions such as buying, selling, exercising, simulating European-style options contracts, calculating Poption prices and calculating arbitrage strategies and returns. Next, we will fulfill items described in the 'Creating Market' section of this paper, to set up an ecosystem. The key projects to develope are the decoupling of components, distribution, development of basic tools and front-end price routing.

7 Disclaimer

This paper is for general informational purposes only. It does not constitute a recommendation or opinion to buy or sale any investment and should not be used in the evaluation of any investment decision. The contents of this document are subject to change or update without notice.

References

[1] Hayden Adams, Noah Zinsmeister, and Dan Robinson. "Uniswap v2 Core". In: (2020). [Online; accessed 22-Feb-2022]. URL: https://uniswap.org/whitepaper.pdf.

[2] Hayden Adams et al. "Uniswap v3 Core". In: (2021). [Online; accessed 22-Feb-2022]. URL: https://uniswap.org/whitepaper-v3.pdf.

[3] DK Aggarwal. "What happens when stock prices shoot up because of Gamma Squeeze". In: *The Economic Times* (May 2021). [Online; accessed 22-Feb-2022]. URL: https://economictimes.indiatimes.com/markets/stocks/news/what-happens-when-stock-prices-shoot-up-because-of-gamma-squeeze/articleshow/83059269.cms?utm_source=contentofinterest&utm_medium=text&utm_campaign=cppst.

[4] Guillermo Angeris and Tarun Chitra. "Improved Price Oracles: Constant Function Market Makers". In: *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies* (Oct. 2020). DOI: 10.1145/3419614.3423251. URL: http://dx.doi.org/10.1145/3419614.3423251.

[5] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. DOI: 10.1017/CBO9780511804441.

[6] Sean Dawson et al. "Lyra". In: (2021). [Online; accessed 22-Feb-2022]. URL: https://www.lyra.finance/files/whitepaper.pdf.

[7] Robin Hanson. "Combinatorial Information Market Design". In: *Information Systems Frontiers* 5.1 (Jan. 2003), pp. 107–119. ISSN: 1572-9419. DOI: 10.1023/A:1022058209073. URL: https://doi.org/10.1023/A:1022058209073.

[8] John C. Hull. *Options, futures, and other derivatives*. 6. ed., Pearson internat. ed. Upper Saddle River, NJ [u.a.]: Pearson Prentice Hall, 2006. XXII, 789. ISBN: 978-0-13-197705-1. URL: http://gso.gbv.de/DB=2.1/CMD?ACT=SRCHA&SRT=YOP&IKT=1016&TRM=ppn+563580607&sourceid=fbw_bibsonomy.

[9] *Introduction to Ribbon*. [Online; accessed 22-Feb-2022]. URL: https://docs.ribbon.finance/.

[10] Antonio Juliano. "dYdX: A Standard for Decentralized Margin Trading and Derivatives". In: (2017). [Online; accessed 22-Feb-2022]. URL: https://whitepaper.dydx.exchange/.

[11] Zubin Koticha. "Convexity Protocol : Building a Generalized Liquid Options Protocol in DeFi". In: (2019). [Online; accessed 22-Feb-2022]. URL: https://drive.google.com/file/d/1YsrGBUpZoPvFLtcwkEYkxNhogWCU772D/view.

[12] Wikipedia contributors. *Black–Scholes model — Wikipedia, The Free Encyclopedia*. [Online; accessed 22-Feb-2022]. 2022. URL: https://en.wikipedia.org/w/index.php?title=Black%E2%80%93Scholes_model&oldid=1071176882.

[13] Wikipedia contributors. *Short squeeze — Wikipedia, The Free Encyclopedia*. [Online; accessed 22-Feb-2022]. 2022. URL: https://en.wikipedia.org/w/index.php?title=Short_squeeze&oldid=1069107423.