ArithFi: A New DeFi Paradigm Based on SCP

Contents

ArithFi: A New DeFi Paradigm Based on SCP

ArithFi-DAO

2023-11-11

Liquidity is the core of on-chain applications. To address this, traditional decentralized finance (DeFi) projects have employed conventional order book models and automated market maker (AMM) models. However, these models have not provided ideal solutions and have failed to integrate all financial services within a single protocol, leading to resource waste and inefficient operations. This article proposes an innovative paradigm: the ArithFi protocol, which introduces the concept of System Contract Counterparty (SCP) and on-chain monetary units, fundamentally addressing the liquidity and uniformity issues in DeFi. It is applicable for developing various financial products and establishing economic relationships that lock in off-chain activities.

1. History of DeFi

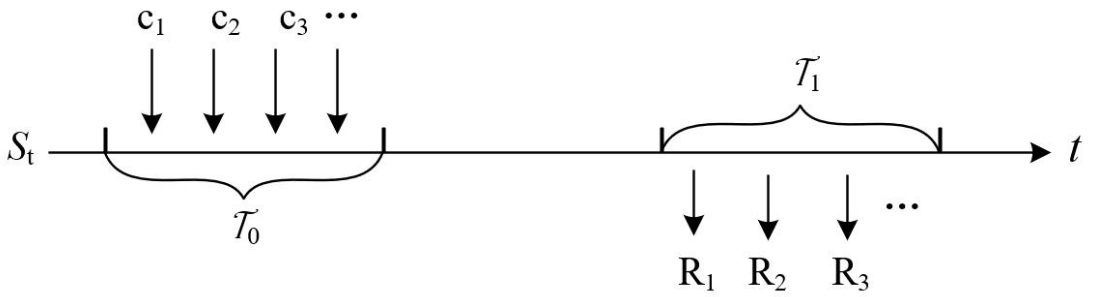
The development of DeFi can be traced back to early on-chain trading and peer-to-peer lending, which gained attention around 2017. However, due to the high cost of on-chain matching and the lack of decentralized oracles, these projects did not experience sustained development. Instead, projects like Uniswap, based on the AMM mechanism, and lending protocols like Compound and MakerDao, based on liquidity pools and asset pricing, rapidly rose to prominence, leading the DeFi trend. These projects achieved better results in addressing on-chain liquidity scarcity.

However, both AMM and liquidity pools sacrifice the flexibility of the selling side: sellers must fix their trading strategy and bear the risk of market fluctuations. Once the market favors sellers, buyers may exit the trade, and when arbitrage opportunities arise, buyers flock in. Throughout this process, sellers lack the right to choose and can only rely on mining subsidies and commission or interest rate balances under the law of large numbers. While this design temporarily alleviates liquidity issues, in the long run, it faces problems such as resource waste, susceptibility to arbitrage, and non-sharing of TVL.

A decentralized architecture should not require asymmetric sacrifices from the selling side. Instead, we propose the System Contract Counterparty (SCP) model, where transactions occur not between dispersed users but directly between users and contracts. This concept aligns with the essence of decentralized blockchain gaming, where all participants fairly compete under the same rules. Users only need to pay tokens to the system to obtain the desired financial products, and the returns of financial products are settled through the issuance of system tokens. This new approach not only fundamentally changes our understanding of traditional financial trading models but also ensures the high composability of DeFi and achieves unified programming capabilities within the same framework.

2. ArithFi Principles

All financial services and products are fundamentally a form of transaction involving the exchange between the current cost paid and the expected future return, depicted as follows:



S_t represents the information flow of price or interest rate, R_i represents the revenue stream, c_i represents the expenditure stream, $\mathcal{T}_0$ is the time domain for expenditure, $\mathcal{T}_1$ is the time domain for revenue.

The core principle of ArithFi is based on a specific discounting algorithm that equalizes the future cash flows with the current outflows in value (or slightly lower, promoting deflation). In this way, each expected cash flow is matched with a specific outflow, and both are settled using the ATF token. If we consider the expected cash flow as a financial product, the corresponding outflow can be seen as the present value or cost of that product. For simplicity, we will uniformly refer to this outflow as the cost. The entire process can be summarized as follows:

$$\sum_{\mathcal{T}_1} E\left[e^{-rt_i} R_i / \mathcal{F}_0\right] \leqslant \sum_{\mathcal{T}_0} E\left[e^{-rt_i} C_i / \mathcal{F}_0\right]$$

Note: r is the discount rate, R_i is the revenue stream, c_i is the expenditure stream, $\mathcal{F}_0$ is the information set at the time of the transaction, $\mathcal{T}_0$ is the time domain for expenditure, $\mathcal{T}_1$ is the time domain for revenue.

Financial products can be considered as composed of a linear combination of Basic Revenue Functions (BRF), each with a corresponding discounting function known as Basic Discount Functions (BDF). In this way, the cost of a financial product is essentially the result of the linear combination of these basic discounting functions. Therefore, we can construct the BRF and its corresponding BDF into a programmable module: Mfunction. Using this module, any financial product can be developed and constructed. Here, the basic revenue functions are similar to a computer's instruction set, and the basic discounting functions are akin to the costs of these instructions or a concept similar to gas fees in the Ethereum Virtual Machine (EVM). The difference is that here, the "gas" is paid in ATF tokens, and these instructions actually generate ATF tokens. The method for calculating financial products and their costs is demonstrated below:

$$P = x_1 \cdot BRF_1 + X_2 BRF_2 + \cdots = \boldsymbol{BRF} \cdot \boldsymbol{X}^T$$

$$C(P) = x_1 BDF_1 + X_2 BDF_2 + \cdots = \boldsymbol{BDF} \cdot \boldsymbol{X}^T$$

P is a financial product, $C(P)$ is its cost

X is the BRF of P , representing the discount function BDF of BRF

Here: $E\left[e^{-rt_i} BRF_i / \mathcal{F}_0\right] \leq E\left[e^{-rt_i} BDF_i / \mathcal{F}_0\right]$

Using the Mfunction module, developers can create a variety of financial products, including options, perpetual contracts, leveraged trading, swap protocols, standard transactions, and lending, among others. In fact, the flexibility of the Mfunction module enables the construction and implementation of almost all types of financial products.

3. ATF: Issuance, Settlement, and Pricing

ATF is a decentralized currency unit issued by the ArithFi protocol, abbreviated as a Decentralized Currency Unit (DCU). The initial issuance of ATF will not exceed one billion units. In the ArithFi system, ATF serves as the sole currency unit, used for both user payments and received rewards.

For instance, if a certain condition is met in the future, and you are entitled to receive 300 ATF, you would need to pay 50 ATF now. These 50 ATF will be burned, and after 300,000 blocks, you will receive 300 ATF, which are newly issued by the system.

This mechanism implies that all ATF holders collectively bear the risks and rewards associated with ATF issuance or destruction. They simultaneously participate in the supply-demand balance of ATF in the secondary market. Those in need of ATF include users purchasing on-chain financial products and investors investing in ATF. The supply of ATF is determined by the initial issuance plus newly issued ATF calculated by the ArithFi protocol. The equilibrium in the exchange is achieved through prices.

Using a unified valuation unit has a significant advantage: by continually enhancing ATF liquidity, we can provide solutions for all financial services without the need to create numerous tokens. Whether it's trading, lending, or derivative transactions, they can all be conducted using ATF as the valuation, payment, and settlement unit.

According to $E\left[e^{-rt} BRF / \mathcal{F}_0\right] \leq E\left[e^{-rt} BDF / \mathcal{F}_0\right]$

the total supply G_t satisfies $EG_{t_2} \leqslant EG_{t_1}, t_2 \geqslant t_1$

The total demand D_t is determined by transactional needs.

P_t is determined by the equilibrium of (D_t, G_t)

Considering the growth nature of demand and the long-term deflationary characteristics of supply, there is a logical basis for the continuous rise of P_t

ATF, as defined, combines with the ArithFi protocol to become an on-chain universal currency with specific use-case scenarios. It achieves features that Bitcoin (BTC) and Ethereum (ETH) have not been able to realize: BTC, despite having a fixed issuance, lacks on-chain use cases, while ETH, although accompanied by various applications as gas, follows a fixed algorithm for issuance and does not increase based on specific scenarios. In contrast, ATF ensures transaction settlement in every scenario, aligning with the vision of many economists for a fully decentralized currency, marking a significant advancement beyond BTC and ETH.

4. Oracle

NEST Oracle stands out in the market as the only fully decentralized oracle. Its core challenge lies in designing a decentralized game mechanism to ensure that off-chain price feeds accurately reflect on-chain prices while minimizing deviations between the two. Through a series of innovative mechanisms, including quote mining, bidirectional options, verification cycles, price chains, and beta coefficients, NEST Oracle cleverly addresses this issue, marking a pinnacle in design.

The price data series provided by NEST does not alter the fundamental distribution of asset prices but rather resembles discrete sample collection. The design of this mechanism determines the bias and density of quotes, depending on the depth of arbitrage markets and the price of NEST tokens. Overall, NEST Oracle offers an effective decentralized solution, preserving the fundamental properties of prices.

In the architecture of ArithFi, we prefer using price data from efficient markets. Therefore, we have chosen highly liquid assets like BTC and ETH as the basis for pricing. The fundamental price model we adopt is Geometric Brownian Motion (GBM). Simultaneously, considering the deviations in actual prices and the discreteness of time, we adjust prices based on the GBM model, known as the so-called k-factor correction. A feasible k-factor is as follows: [Insert the k-factor values].

$$K = (0.00002 * T + 4 * \sigma) * \gamma(\sigma)$$

σ is second-level volatility

T represents the time delay: T = (Block height of successful packaging - Block height of the most recent valid NEST price) * timespan

$$\gamma = \begin{cases} 1 & \sigma \le 0.0003 \\ 1.5 & 0.0003 < \sigma \le 0.0005 \\ 2 & \sigma > 0.0005 \end{cases}$$

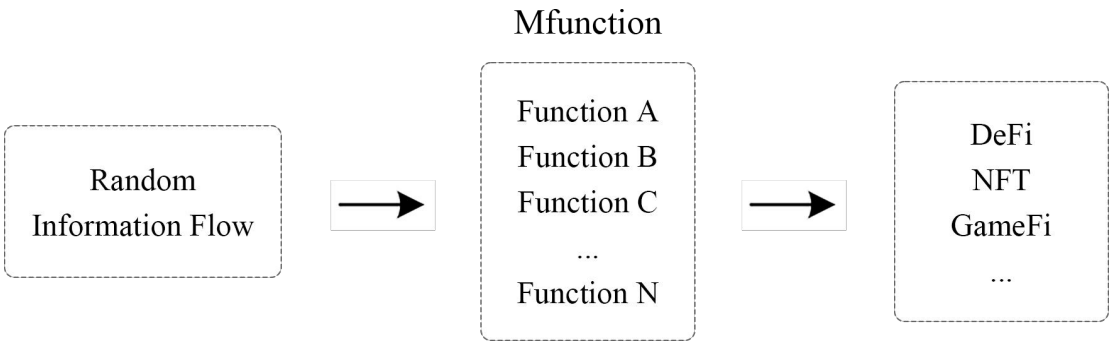
5. Time domain

When representing the time domain, we use $\mathcal{T}_i$ and primarily distinguish it into specific moments and time intervals. A moment can be deterministic, such as a particular point in time, or

indeterministic, triggered by random events (e.g., so-called stopping times). In the financial domain, time intervals are often used to determine averages over a period or the occurrence of a stopping time. Although the recording of time on the blockchain is inherently discrete, these discrete differences can often be neglected over longer time scales. For shorter time periods, we can adjust for this discreteness by applying a k-factor. Therefore, when dealing with problems, we can use the concept of continuous time intervals as an approximation.

6. Mfunction

We have conceptualized all financial products and services as an interchange process between revenue flows and expenditure flows. In this process, the revenue flow consists of a linear combination of fundamental revenue functions. Therefore, the key step in developing any financial product is to determine the linear combination of fundamental revenue functions. Once this combination is identified, we can calculate the cost (present value) of the product through a linear combination of the corresponding discount functions. The process of this linear combination is very similar to the methods used in computer programming. It is for this reason that we have named this module "Mfunction." Its operation can be represented by the following schematic diagram:



In this framework, any financial product can be implemented by writing a piece of Mfunction code. Thus, the composability of DeFi transforms into a unified framework for program design and execution, simplifying the understanding of complexity and easing the challenges of risk management.

Mfunction constitutes the foundation of financial products, combining fundamental revenue functions (BRF) and discount functions (BDF). The fundamental revenue function may represent a deterministic value (e.g., obtaining 1000 ATF in block 13678933) or, when combined with prices provided by the NEST oracle, it may become a random variable. The basic types we handle include deterministic values, random variables derived from the NEST oracle, and purely probabilistic random variables. Each type can be expressed using polynomials, exponential functions, logarithmic functions, absolute value functions, max/min functions, and definite integral operators. As for the discount functions, they encompass the cumulative distribution function (CDF) along with polynomial functions, exponential functions, logarithmic functions, etc. Considering that an extensive variety of revenue function types is not necessarily required in practical applications and to reduce computational complexity, we have opted for a relatively

simple function list, which can be progressively expanded and refined as needed.

As mentioned earlier, the fundamental revenue functions can be regarded as the basic instruction set of Mfunction, with each financial product analogous to a program composed of these instructions.

7. Discount Rates and Interest Rate Oracle

In principle, the discount rate is used to represent the risk-free rate of return in the on-chain world. We have various options to determine this rate, such as using the Ethereum Proof of Stake (POS) yield rate or leveraging the interest rate provided by decentralized interest rate oracles. One possible design is based on the annual issuance of ATF, where anyone holding and staking ATF can participate in the allocation of this newly issued supply. However, such discount rate settings are based on traditional centralized financial perspectives. In a decentralized world, to reflect the deflationary characteristics of ATF issuance and ensure the stable growth of ATF value, we can choose to set a relatively low discount rate, or even set it to zero.

8. Unit of Account Transformation

In the ArithFi system, if it is necessary to use fiat currency or Ethereum (ETH) as the unit of account, we can simply introduce the exchange rate between ATF and USDT or ATF and ETH. This exchange rate can be obtained through the NEST oracle. With sufficient ATF liquidity, even if the settlement of individual financial products has a minor impact on ATF prices, the introduction of ATF into financial products is essentially no different from traditional financial products. In such cases, methods based on risk-neutral pricing theory can be effectively used to calculate discount functions. These financial products can be applied to hedging strategies or portfolio management.

9. Financial Product Development

In the ArithFi system, the development of financial products is similar to the process of writing smart contracts. Developers need to determine the target return and build a vector based on the fundamental revenue function (BRF) as the basis, representing the financial product being developed. This vector multiplied by the corresponding discount function (BDF) represents the cost of the financial product. Users only need to pay this cost within the time domain to obtain the corresponding financial product. Over time, users will receive newly issued ATF from the ArithFi contract, the quantity of which is equivalent to the product of this vector and BRF.

The entire development process is like writing a standard smart contract code, a systematic operation. This means that any desired financial product can be programmed using ArithFi's Mfunction, ensuring uniformity of products. Additionally, developers do not need to operate their own token liquidity; they only need to ensure that ATF itself has sufficient liquidity.

10. Application Examples

The application scope of the ArithFi protocol is exceptionally broad, encompassing nearly all areas of financial services, including various transaction structures (such as peer-to-peer, many-to-many, etc.), while also capable of capturing various off-chain economic relationships. It represents a significant milestone in the history of blockchain development.

Options and Option Tokens

In ArithFi, issuing options becomes extremely straightforward. By inputting the expiration date and strike price, one can obtain a call or put option. Its cost is determined by the discount function, but without introducing ATF prices, the formula is not risk-neutral. When ATF prices are introduced, it aligns with traditional options; however, the interaction process is significantly simplified, eliminating the need to consider complex matching processes.

An enhanced model transforms options into tokenized forms. Under specific expiration dates and strike prices, it is the same token regardless of when it is issued. The advantage of this model is that it allows traditional derivative exchanges to focus solely on secondary market trading without concern for issuance and settlement issues.

Perpetual Contracts, Leveraged Trading, and Leveraged Tokens

Perpetual contracts and leveraged trading are greatly simplified in ArithFi. They are based on the fundamental revenue function of dynamic settlement and can also be developed into a leveraged token model that dynamically adjusts balances, similar to current algorithmic stablecoins.

Trading, Price Tokens, and Stablecoins

In ArithFi, a native asset can be equivalent to a price token multiplied by the ATF unit of account, essentially splitting the asset into dynamic price and fixed settlement units. This model is effective only in a fully decentralized world. Therefore, trading involves exchanging various price tokens with ATF or vice versa, or swapping native assets for price tokens in a 1:1 exchange (with a slight deviation caused by the oracle's bias). Similarly, stablecoins anchored to fiat currencies, such as USDT, are considered price tokens with a USD price.

Index Tokens and Logarithmic Tokens

Index tokens are a new concept that reflects the ratio of price volatility exponentially onto returns. Compared to leveraged tokens, index tokens do not require liquidation, grow faster, can be transferred mutually, and the same address can freely stack them. For example, when the price doubles, an index token with the base of 'e' can grow 7.4 times, and doubling results in a 20-fold increase.

Revenue Swaps

Future revenue swaps essentially involve the exchange of costs since they represent the discounting of future revenue flows.

Lending

The lending process becomes simpler in ArithFi. Users can collateralize assets recognized by

ArithFi contracts to receive corresponding ATF and can retrieve the collateralized assets upon repayment. If the liquidation threshold is reached, liquidation is triggered. Key parameters for this process include the liquidation threshold, collateral ratio, and interest rate.

Insurance

Price insurance can be designed based on the tail risk characteristics of specific events, exchanging potential tail losses with premiums.

Probability Tokens

In ArithFi, a special type of token can be created with the characteristic of winning ATF at a predetermined probability at a given time point. For example, a user holding a "one-tenth probability token" will have a one-tenth chance of winning 10 ATF.

NFTApplications

ArithFi allows for the economic relationships of any off-chain game or NFT to be locked based on ATF, ensuring that all game assets can correspond to some probability token or the aforementioned derivatives. This means that regardless of the game to which game assets belong, their corresponding NFTs can be exchanged on the ArithFi platform, ensuring consistency in asset value in the gaming world, even if the game itself no longer exists.

Multilateral Transactions

ArithFi also allows for the design of multi-party transaction contracts involving two or more participants. In such contracts, parties can pay a certain amount of ATF at the current moment and receive corresponding ATF returns in the future based on contract terms. This structure enables ArithFi to intervene and regulate allocations among participants, creating a multi-party competitive and strategic trading environment.

11. Summary

ArithFi introduces an innovative paradigm by treating financial products as a programming expression of basic discount functions, where the cost is equivalent to the expense of calling these functions. This is somewhat similar to the working principle of the Ethereum Virtual Machine (EVM), but the distinction lies in the fact that the economic relationships of Mfunction are generated internally within the system. This paradigm can cover the vast majority of financial products and services, providing the ability for instantaneous purchase and settlement with unlimited liquidity. In this system, there is no longer a need for market makers, concerns about margin calls or forced liquidations, and risks associated with settlement failures.

As long as ATF has sufficient liquidity, simulating traditional financial markets becomes exceptionally simple, and the capabilities of ArithFi become very powerful. Additionally, due to the simplification of the issuance and settlement processes, traditional derivative exchanges can focus on operating the secondary market, significantly reducing their operational costs. ArithFi also has the potential to serve as the foundational consistent variable for building the currently popular metaverse, allowing it to lock economic relationships across different gaming platforms, showcasing a wide range of application prospects.