Premia v3 Specification

Froggie, Enso, Wolfy, Tolga Guceri, and Magnus Wiese

research@premia.finance
research@openhedge.com
Premia Finance

January 13, 2023

Abstract

PREMIA V3 is a non-custodial options settlement engine and automated market maker implemented for the Ethereum Virtual Machine. This version of the protocol implements a base layer exchange that enables the permission-less creation of option pools. Each pool utilises concentrated liquidity, partial collateralisation, prorata fee growth for liquidity providers, and integrations for strategic vaults and quote systems. These innovations improve the capital efficiency, composability, and sustainability of the protocol.

1. Introduction

In recent years, automated market makers (AMMs) have catalysed a significant paradigm shift in how asset exchanges are created and utilised in decentralised finance (DeFi) [8], providing benefits such as deeper liquidity, self-custody of assets, and transparent market data. AMMs are non-custodial asset exchanges that enable users to either provide liquidity to make a market or to buy and sell assets from the market, at a price automatically determined by the supply and demand on the exchange. The on-demand characteristic of automated exchanges enables liquidity providers (LPs) to passively participate in fee collection from market-making, while traders get access to counterparties for their trades with continuous availability.

1.1. Related Works

While on-chain scaling and automated markets are not a new concepts, the last few years have ushered in a plethora of innovations that have ultimately made these types of protocols significantly more efficient.

1.1.1. OTHER PROTOCOLS

Uniswap v3 introduced the concept of concentrated liquidity for AMMs [1] and since its inception, the idea has been explored by multiple other protocols such as Curve v2 [4] spurring the creation of more constant function market makers (CFMMs) [2]. The exchange uses range orders and tick-based pricing in order to optimise traversal through the price domain. This innovation, among others, enabled Uniswap v3 to attract more professional market-makers to their spot exchange than previous versions of the protocol. The increased liquidity in concentrated ranges results in lower price impact and higher liquidity provider (LP) fee revenue for trades of similar size than on non-concentrated AMMs.

Option protocols, such as Ribbon (built on top of Opyn) and Lyra [3], have become incrementally successful as specialised option vaults. These structures have lowered the barrier to entry for many participants who desire passive investment products and yield-based strategies. Vaults have also been able to temporarily solve scaling issues that made on-chain order books impractical in previous years.

1.1.2. PREMIA V2

PREMIA V2 [5], was originally built as an AMM focused on buyers vs. sellers, as opposed to the more common maker vs. taker model. LPs are fully collateralised option sellers, by default, but can not select specific options on the volatility surface to underwrite, instead, they provide liquidity to the entire surface. Without the ability to create concentrated liquidity positions, it is necessary for trade prices to be driven by more than just order size. Off-chain implied volatility (IV) oracles are combined with on-chain supply vs. demand flows to accurately price each market.

Addressing all the functionality shortcomings from the previous design, PREMIA V3 is a complete rework of the protocol from the ground up. The latest in AMM architecture is used to enable the most capital-efficient options exchange currently available in DeFi. Particular attention was paid not only to functionality, but additionally composability and scalability of the base layer exchange.

2. Protocol Overview

The major aim of this paper is to construct liquidity pools for any call or put option where an LP (maker) can provide liquidity at an automated market price. A call (put) option is a financial asset that gives the option holder the right, but not the obligation, to buy (sell) an underlying asset at a pre-determined date T, the *maturity date*, and a pre-determined price K, known as the option's *strike*. Every pool is designed to represent a single option (K, T).

PREMIA V3 enables the creation of European-style options markets for any asset pair with an on-chain spot price oracle. LPs can select, in advance, exactly which option(s) they would like to trade and the price range(s) they are willing to trade within. The exchange behaves similarly to a central-limit order book for options, while retaining the benefits of an advanced, concentrated AMM. Contrary to conventional CFMMs [1], the AMM introduced in this paper features linear pricing. PREMIA V3 is meant to be a modular and layered protocol to create maximum composability and upgradability on top of the primitive exchange layer.

The main innovation of PREMIA V3 is the split-accounting system (uni-directional, concentrated liquidity), which follows traditional AMM, pro-rata accounting mechanics. In order to successfully use an AMM architecture for options, both long and short positions must be attainable for individual LPs, while efficiently maintaining global and position liquidity state within the pool.

2.1. Concentrated Liquidity

LPs are able to create positions in specific option pools as concentrated range orders with defined upper and lower price bounds. Each range order belongs to a single pool for a specific option (combination of strike price, maturity date, and option type). This enables active traders with high conviction to maximise fee collection from highly capital efficient, concentrated orders, while passive LPs with less conviction can still earn trading fees in a wider, less capitally efficient range. LP positions and takers' option positions are represented as fungible ERC-1155 tokens. Both can be transferred between users and traded on third-party exchanges.

2.2. Partial Collateralisation

Margin architecture is a core requirement for any full-scale options market. Margin enables partial collateralisation of options, which facilitates greater amounts of liquidity and in turn produces more efficiently-priced assets. In order to avoid a fractional reserve system, a lending market must be used to fully collateralise option positions at the exchange layer, while enabling sellers to provide a smaller portion of collateral on margin. This ensures that in even the most extreme market conditions, the exchange is able to maintain solvency and effectively eliminate counter-party risk.

2.3. Transaction Fees & Liquidity Mining

Transaction fees are paid by traders taking liquidity (takers) and split between LPs (makers) and protocol stakeholders at a rate determined by governance. To retain higher composability, transaction fees are accumulated separately and not compounded back into LP positions. Users can stake PREMIA tokens (to receive vxPREMIA) on-chain to collect transaction fees and acquire voting rights to direct liquidity mining rewards to specific token pairs. Liquidity mining rewards are paid in PREMIA token to LPs, proportional to the size, location on the volatility surface, and price competitiveness of each order.

2.4. Vaults & OTC Liquidity

One of the many challenges with options markets is liquidity fragmentation across strikes and maturities. The over-the-counter (OTC) quote system, built into the base exchange layer, enables any vault or market-maker to provide fillable option quotes to users (on-chain or off-chain), minimising the fragmentation of liquidity across the exchange. This provides just-in-time (JIT) liquidity mechanisms to vaults and market-makers, enabling highly capital efficient market-making for both passive and active users. In addition to OTC liquidity, vault creators can utilise standard range orders to fulfill automated strategies on-chain.

3. Market Structure

PREMIA V3 enables the permissionless creation of option markets for any pair of ERC-20 assets with a supported *spot price* oracle on-chain. An oracle is a smart contract that provides an API for other smart contracts to query data. This is required to accurately determine the spot price at expiration of each option.

3.1. Pool Initialisation

Generally speaking, initialising a pool for a new option market (K, T) is a permissionless process. However, there is a balance between permissionless creation of markets for any strike (K) and maturity (T), and preventing liquidity fragmentation over too many options.

The *strike interval* for an asset pair (eg. ETH/USDC) determines the granularity of strike prices that can be used to initialise a new option market for that pair. Denominated in the numéraire (eg. USDC), strike intervals are algorithmically set at the protocol level to a log-rounded number, based on the current market price of the asset. The strike interval for an asset pair can be updated by governance processes.

All newly created option markets must expire at *8 AM UTC*, with the additional stipulation that options with maturities over 2 days must expire on a *Friday*, and options with maturities over 30 days must expire on the *last Friday* of the calendar month. Daily options can only be created with expirations less than 3 days, and weekly options can only be created with expirations less than 30 days. The maximum time to maturity is currently set to one year, though this can be updated by governance.

In order to prioritise the creation of options with high general utility, an option's *moneyness* and *duration* are used to determine an initialisation fee that is paid directly to the protocol. Ceteris paribus, at-the-money (ATM) options require the lowest fees to create, and near-dated options are cheaper than far-dated options. Fee discounts are provided for *each* option pool yet to be initialised, which has nearby initialised options with a similar strike (K) or expiration (T). The fee is designed to become negligible as a market matures.

3.2. Price Normalisation

The PREMIA V3 invariant requires price intervals that are range-bound, i.e. have a known lower- (zero) and upper-bound (maximum spot price). To circumvent a hard-coded cap, call option prices are denominated in the underlying. This normalises the price range of call options to the price interval $[0, 1]$. Hence, at time $t \geq 0$ a call option with price p quoted in the underlying (e.g. ETH) will therefore be offered at the exchange for the *normalised price*, pS_t^{-1}, where S_t is the price of the underlying at current time t. For put options, the price is simply quoted in the numéraire (e.g. USDC) and normalised using the strike K to similarly bound the price range to $[0, 1]$. Note that all prices referenced throughout the remainder of the paper are normalised as defined.

3.3. Ticks

A tick is a data structure used to optimise storage and traversal of liquidity within distinct price ranges. Each tick maps to a specific price within the price space $[0, 1]$ and stores data necessary for concentrated range orders to reconcile pool state changes over time. This enables the pool to find the exact amount of liquidity necessary to traverse between two prices in the pool, to find the next market price given some trade quantity, and to calculate the fees earned by LPs in a price range over a period of time.

The number of ticks, N, used to define the tick space, controls how granular range orders can be placed and can be updated by governance. Furthermore, the *minimum tick distance* (defined as the inverse of N) influences the maximum gas-intensity, i.e. the upper-bound of Ethereum Virtual Machine (EVM) network fees that need to be paid to verify each trade in the pool. Ticks are spaced equidistantly within the price range, thus, the tick space is defined as $\mathcal{T} = \{\frac{i}{N}\}_{i=1}^{N}$.

3.4. Range Orders

Range orders are customised orders that enable LPs to deposit liquidity within a price range defined by a lower (τ_l) and an upper tick (τ_u). Orders can either be placed *above* or below the current market price, inclusively. Orders that span the *minimum* tick distance can replicate limit-orders similar to a limit order-book, while orders spanning multiple ticks replicate orders placed linearly over multiple adjacent prices. Using multiple range orders, LPs can replicate complex, non-linear liquidity distributions.

Range orders can be thought of as self-replacing limit-orders, where they initially act as a standard limit-order with maker rebates, however, once filled, an equal and opposite limit-order will be placed on the exchange at the prior limit price, and this process repeats if the replacement order is filled. This enables market makers to continually collect maker fees with a single order as price traverses up and down continuously through the range.

When liquidity is deposited above the market price, the order can be *initially* viewed as *ask-side* liquidity. Similarly, deposits below the market price can be *initially* viewed as *bid-side* liquidity. Each of these order types acts as a way of liquidity provisioning and therefore is rewarded with a *maker fee* when filled. LPs can simulate standard limit-orders by removing liquidity when a desired exposure is obtained.

Note: Market orders cannot be reflected as LP positions within the AMM. Rather, users must act as a taker and will have to pay a taker fee to trade against the pool.

4. Split-User Accounting

Premia v3 employs a split accounting system to track long and short exposures of market participants. Traders can buy and sell option contracts through the liquidity provided by LPs. An LP provides liquidity by supplying the exchange with collateral or option contracts. Collateral is provided for call option pools in the underlying to ensure full collateralisation, whereas for put option pools, collateral is provided in the numéraire, such as USDC, requiring K tokens of the numéraire per underwritten option contract to attain full collateralisation. Option contracts can be deposited in the form of long and short options.

For an LP to gain different exposures, liquidity can be provided through a range order above or below the market price. Four different range orders are supported by the exchange:

- **Sell-with-collateral.** Underwrite / sell option contracts through the provision of collateral.
- **Buy-with-shorts.** Provide bid-side liquidity by covering short position by buying option contracts from traders.
- **Buy-with-collateral.** Buy option contracts by using collateral.
- **Sell-with-longs.** Provide ask-side liquidity through selling long option contracts.

The core of the exchange is a linear pricing system. This is in stark contrast to well-known exchanges such as Uniswap where a trading function governs the exchange of assets resulting in non-linear pricing. For the exchange to satisfy the linear relationship between assets exchanged and price, range orders have to provide liquidity (*option contracts*) linearly within their range. Following, the exchange of collateral and contracts through the provision of liquidity are discussed for each range order below.

Sell-with-collateral Collateral is deposited above the market price to underwrite call or put options allowing an LP to gain short exposure after full traversal of the range order. An LP that deposits c units of collateral can underwrite a total of c ($\frac{c}{K}$) fully collateralised call (put) options respectively.[1] Whenever the market price is below (above) the lower (upper) tick of the range order the range order solely consists of c units of collateral (c [$\frac{c}{K}$] short call [put] options). During traversal the liquidity is a mix of collateral[2] and short options where the position's composition depends on the market price $p \in [0, 1]$

$$\begin{aligned} \text{collateral}(p) &= c\,(1-\nu(p)) + c\,\omega(p) \\ \text{short}(p) &= \nu(p) \begin{cases} c & \text{if call option} \\ \frac{c}{K} & \text{if put option} \end{cases} \end{aligned} \quad (1)$$

[1]This is due to call (put) option pools being denominated in the underlying (numéraire).

[2]Note that the collateral function in equation (1) consists of the collateral used to underwrite options as well as the option premiums received from selling the options.

where the function

$$\nu(p) = \begin{cases} 0 & \text{if } p \in [0, \tau_l) \\ \frac{p-\tau_l}{\tau_u-\tau_l} & \text{if } p \in [\tau_l, \tau_u) \\ 1 & \text{if } p \in [\tau_u, 1] \end{cases}$$

is a piece-wise linear function which determines the growth of options held by the LP and the relative decline of collateral from underwriting options. Furthermore,

$$\omega(p) = \nu(p) \left(\nu(p) \frac{\tau_u - \tau_l}{2} + \tau_l \right)$$

determines the revenue generated per unit of collateral provided.

The liquidity composition as a function of price is visualised for a range order with lower tick price $\tau_l = 0.2$, upper tick price $\tau_u = 0.8$ and $c = 2$ units of collateral at time of deposit in Figure 1. Note that after traversal the position holds 1 unit of collateral since the *average price* $\mu := \frac{\tau_l+\tau_u}{2}$ is 0.5.

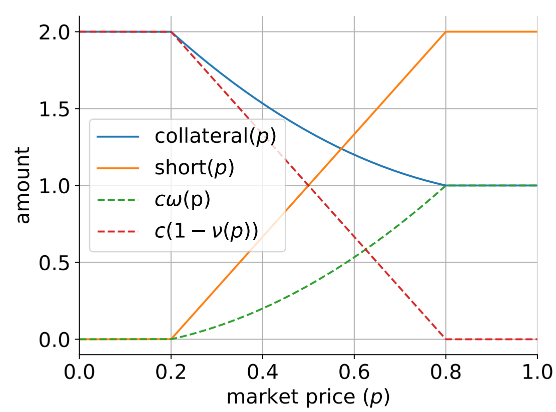

Figure 1: Liquidity composition of a sell-with-collateral / buy-with-shorts order type in a call option pool as a function of the current market price p. The dashed red illustrates the collateral used to underwrite options, whereas the dashed green line depicts the option premiums from underwriting options. The sum of both functions returns the amount of collateral held by the LP.

Buy-with-shorts A buy-with-shorts order type allows LPs to buy-to-close their short exposure by offering bid-side liquidity. This allows takers to sell long or open short option contracts. At time of deposit short contracts (d) are deposited below the market price. Furthermore, $d\mu$ ($dK\mu$) units of collateral need to be deposited in a call (put) option pool to cover the option premiums paid to takers. The amount of collateral and shorts held for any market price

$p \in [0,1]$ is governed by the equations

$$\begin{aligned} \text{collateral}(p) &= \tilde{d}\,(1-\nu(p)) + \tilde{d}\,\omega(p) \\ \text{short}(p) &= d\,\nu(p) \end{aligned} \quad (2)$$

where $\tilde{d}$ is d (dK) for call (put) option pools. Observe that equations (1) and (2) are very similar since a sell-with-collateral order can be regarded as a buy-with-collateral order after full traversal and vice versa. Equations (2) are visualised in Figure 1 for a call option pool for tick prices $\tau_l = 0.2$ and $\tau_u = 0.8$, $d = 2$ short option contracts and $1 (= d\mu)$ initial collateral to cover the option premiums.

Buy-with-collateral A buy-with-collateral order type consists initially only of collateral which is deposited below the market price to provide bid-side liquidity. Let $\tilde{\mu}$ be the unnormalised average price, i.e. μ for call pools and μK for put pools. A position holding initially c units of collateral can acquire in total $c\tilde{\mu}^{-1}$ units of options contracts. The liquidity composition at any price $p \in [0,1]$ is given by

$$\begin{aligned} \text{collateral}(p) &= c\,\omega(p) \\ \text{long}(p) &= c\tilde{\mu}^{-1}(1-\nu(p)) \end{aligned} \quad (3)$$

and exemplified in case of a call pool for the ticks $\tau_l = 0.2$ and $\tau_u = 0.8$ and collateral $c = 1$ in Figure 2.

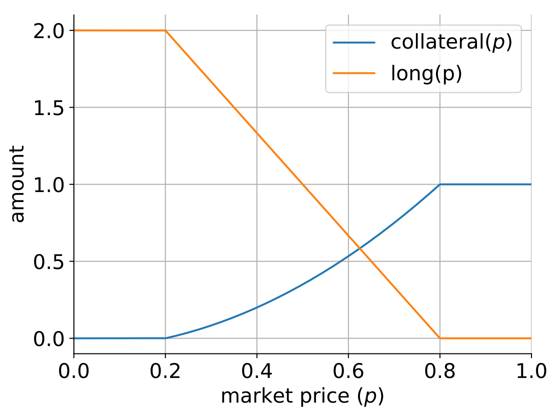

Figure 2: Liquidity composition of a buy-with-collateral / sell-with-longs order type in a call option pool as a function of the current market price p.

Sell-with-longs Long option contracts are deposited above the market price as ask-side liquidity. Selling d long options results in the generation of $d\tilde{\mu}$ units of collateral.

$$\begin{aligned} \text{collateral}(p) &= d\tilde{\mu}\,\omega(p) \\ \text{long}(p) &= d\,(1-\nu(p)) \end{aligned} \quad (4)$$

Again, the similarity of equations (3) and (4) can be observed since a buy-with-collateral order can be considered as a sell-with-longs order after full traversal and vice versa. Equation (4) is illustrated for ticks $\tau_l = 0.2$, $\tau_u = 0.8$ and initial long contracts $d = 2$ for a call option pool in Figure 2.

Straddling the Market Price The range orders types supported by the exchange have to be deposited above or below the market price. There are many ways to configure multiple uni-directional orders to express complex position biases. The most common being a simple *straddle* of the current market price with no position bias. In this case, two range orders need to be created.

5. Pool Accounting

In order to ensure LPs are allocated their pro-rata portion of order flow and trading fees, pool state must be multi-tiered. Each option pool must track storage variables at a *global*, *per-tick*, and *position* level.

5.1. Global State

Liquidity in a pool is distributed uniformly between the lower (τ_l) and upper (τ_u) bounds of each LP's range order and aggregated across all LP range orders on a per-tick basis. Φ represents the amount of global per tick liquidity at each tick index (i). The set of price ticks for a given pool can be broken down into *active* and *non-active* ticks, where active ticks mark a *change* in global per tick liquidity, caused by the addition or subtraction of at least one range order with an endpoint on that tick.

At the *global* (per-pool) level, the following variables are tracked:

Variable Name	Notation
`liquidity_per_tick`	Φ
`price`	p
`tick`	i_c
`fee_growth_global`	ψ^g
`protocol_fees`	ψ^r
`open_positions`	$\{\mathcal{R}\}$
`open_options`	$\{\mathcal{O}\}$

Table 1: Pool-specific global state variables stored on-chain

where `open_positions` and `open_options` are stored as hashmaps of LP positions and taker exposures, respectively.

5.1.1. Liquidity and Price

Since only *active* ticks mark a change in per tick liquidity, the *aggregate pool liquidity* (Λ) between two active ticks

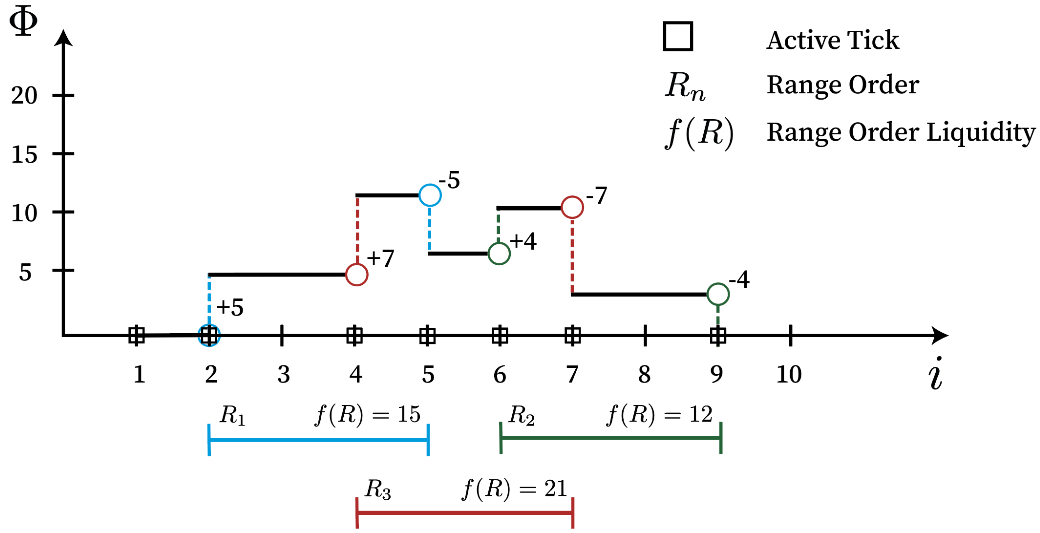

Figure 3: Example of the global liquidity provided per tick across a subset of tick indices ($N = 1000$). The deposited range orders R_1, R_2, and R_3 provide ask-side liquidity, i.e. they are either sell-with-collateral or sell-with-longs order types. The current active tick is tick index 1 and the market price is at tick index 2 and is $0.02 (= \frac{i}{N} = \frac{2}{1000})$. As the price traverses from left to right liquidity per tick is injected / withdrawn. The function $f(R)$ denotes the amount of option contract liquidity that is provided. Note that the first tick index 1 marks an active tick as the first and last ticks are always active tick indices.

can be computed by multiplying `liquidity_per_tick` (Φ) by the number of ticks within an active tick range $[\tau_{i_c}, \tau_{i_c^{++}})$,

$$\Lambda = \Phi \left(i_c^{++} - i_c \right) .$$

Furthermore, since the range orders supported by the exchange provide liquidity linearly a pool's *aggregate ask-* (Λ_{ask}) and *bid-side liquidity* (Λ_{bid}) are also linear between two active consecutive ticks τ_{i_c} and $\tau_{i_c^{++}}$. For a price $p \in [\tau_{i_c}, \tau_{i_c^{++}})$ within the active tick range they can be expressed as

$$\begin{aligned} \Lambda_{\text{ask}}(p) &= \Phi \left(i_c^{++} - Np \right) \\ \Lambda_{\text{bid}}(p) &= \Lambda - \Lambda_{\text{ask}}(p) = \Phi \left(Np - i_c \right) \end{aligned}$$

where N is the total number of ticks in the pool. Consequently, to track the amount of available ask- and bid-side liquidity within an active tick range it is sufficient to keep track of the `liquidity_per_tick` Φ and the `price` p.

5.1.2. Fees

Taker fees are paid by *liquidity takers* (LTs) and split between pool LPs and protocol stakeholders (vxPREMIA holders and the protocol commission pool). The exact fee amount and distribution schedule is subject to change through governance processes. Fees paid by LTs for a trade of size q (the amount of contracts bought or sold) are defined as

$$\chi(q) = \max \left(0.03 \bar{p} q, 0.003 q x \right)$$

where $\bar{p}$ is the unnormalised execution price, and x is the units of collateral employed per-option to collateralise the trade.

Protocol Fees Protocol stakeholders receive a percentage of all taker fees, tracked by the `protocol_fee` (ψ^p) variable. The proportion of fees collected by the protocol (ρ) for each trade will be initialised at 0.5 and may be changed through governance. `protocol_fees` are incremented proportionally every time a trade is executed to track total fees collected by the protocol

$$\psi^p \leftarrow \psi^p + \rho \, \chi(q) \, . \tag{5}$$

LP Fees To capture the fees collected by each LP, the pool tracks the variable `fee_growth_global` (ψ^g), which represents the total growth of fees for each unit of per tick liquidity. It could be interpreted as the pro-rata fee share an LP would receive if depositing one unit of liquidity per-tick across the pool's full price range. This monotonically increasing rate is incremented every time a trade is processed,

$$\psi^g \leftarrow \psi^g + (1 - \rho) \frac{\chi(q)}{\Phi} \, . \tag{6}$$

Note that for simplicity it was assumed in (5) and (6) that the trade size did not trigger a tick crossing (see subsubsection 5.2.2). Whenever a trade crosses active tick indices, the trade is split into multiple partial trades. Each partial trade is then designated to the appropriate active tick interval, and ψ^g is incremented multiple times as per (5) and (6).

5.2. Tick-Indexed State

At the *per-tick* level, each *active* tick τ_i stores the following variables:

Variable Name	Notation
liquidity_net_change	$\Delta\Phi$
fee_growth_external	ψ^e
left_tick	τ_{i--}
right_tick	τ_{i++}

Table 2: Tick-indexed state variables stored on-chain

where left_tick and right_tick are the previous and next *active* ticks, respectively.

Doubly-Linked-List For optimization purposes, the active ticks in the pool are stored in a sorted, doubly-linked list. This enables $\mathcal{O}(1)$ insertion and deletion, given the insert position can be calculated off-chain and verified on-chain.

5.2.1. Swapping within a Single Tick

For small trades that do not traverse an active tick range, the execution price can be computed simply as the *arithmetic* mean of the current market price p and the next price $p(q)$ given some trade size q. Neglecting fees, an LT which is buying (selling) q option contracts will pay (receive) $q\frac{p_0+p(q)}{2}$ in option premiums, denominated in the collateral asset.

5.2.2. Crossing Ticks

When a trade is large enough, the left or right active tick index $i_{\text{cross}} \in \{i_c, i_c^{++}\}$ is crossed resulting in a change in global available per tick liquidity Φ. To account for this change, per tick liquidity is updated using the net change in liquidity defined at the crossed tick, $\Delta\Phi(i_{\text{cross}})$, resulting in the update,

$$\Phi \leftarrow \Phi + \Delta\Phi(i_{\text{cross}}) \ .$$

Thus, if $\Delta\Phi(i_{\text{cross}})$ was positive (negative) prior to crossing, more (less) global per tick liquidity will be available after crossing. Furthermore, whenever a tick is crossed, the sign of the tick's net_liquidity_change is flipped such that a consecutive crossing of the same tick (i_{cross}) would result in the opposite net effect, returning the per tick liquidity to its previous state,

$$\Delta\Phi(i_{\text{cross}}) \leftarrow -\Delta\Phi(i_{\text{cross}}) \ .$$

5.2.3. Tracking Fee Growth

The tick-indexed variable fee_growth_external (ψ^e) is used to be able to compute fee_growth_global above and below the tick. Each time the tick is crossed, the previous state of ψ^e is subtracted from ψ^g to establish a new state for ψ^e

$$\psi^e(i_{\text{cross}}) \leftarrow \psi^g - \psi^e(i_{\text{cross}}) \ .$$

This enables the pool to efficiently track fee growth ownership as price traverses through different active price ranges. Given the current tick, i_c, two simple formulas allow computing the fees earned above

$$\psi^a(i; i_c) = \begin{cases} \psi^g - \psi^e(i) & \text{if } i_c \geq i \\ \psi^e(i) & \text{if } i_c < i \end{cases} ,$$

and below the tick

$$\psi^b(i; i_c) = \begin{cases} \psi^e(i) & \text{if } i_c \geq i \\ \psi^g - \psi^e(i) & \text{if } i_c < i \end{cases} .$$

Using the external fee growth values for a pair of ticks (the lower, i_l, and upper tick, i_u, of an LP's price range), one can determine the total fee growth that has occurred over time within the specified tick range, $\psi_t(i_l, i_u)$, by subtracting the above and below rate from the global,

$$\psi(i_l, i_u; i_c) = \psi^g - \psi^a(i_u; i_c) - \psi^b(i_l; i_c) \ .$$

5.3. Position-Indexed State

Each range order consists of a set of variables that determine both the LP's liquidity state and accrued fees. Orders are identified with a user address, lower tick (τ_l), and upper bound (τ_u). The following five variables are tracked for each order:

Variable Name	Notation
size	s
last_fee_growth	ψ^r
claimable_fees	π
order_type	T

Table 3: LP position-indexed state variables stored on-chain

Note that the variable size is sufficient for any order type to derive the amount of collateral and / or derivatives initially deposited. For sell-with- and buy-with-collateral order types size is initialised as the collateral c deposited, normalised by strike for put option pools. Moreover, sell-with-longs and buy-with-shorts order types set size as the amount of contracts initially deposited. Given the `price`, `size` and `order_type` the liquidity composition can be evaluated using equations (1)-(4) from section 4.

5.3.1. LP Fee Growth

The variable `last_fee_growth` (ψ^r) is used to compute the fees owed to an LP. It is set at the time of deposit (t_0) to the total rate of fee growth, over the full history of the pool, for the LP's price range

$$\psi^r \leftarrow \psi_{t_0}(i_l, i_u) \ .$$

If the range order is modified at a future time, $t > t_0$, either by depositing additional liquidity into the range or (partially) withdrawing, the position's `claimable_fees` (π) gets incremented by the amount of fees that are owed to the LP and `last_fee_growth` gets reset to the price range's current fee growth rate. The fees that are owed to the LP are calculated as the difference between the *current* fee growth, $\psi_t(i_l, i_u)$, and the `last_fee_growth`, ψ^r, multiplied by the liquidity-per-tick of the range order, $\phi(R)$, resulting in the update

$$\begin{aligned} \pi &\leftarrow \pi + \phi(R)(\psi(i_l, i_u) - \psi^r) \\ \psi^r &\leftarrow \psi(i_l, i_u) \end{aligned} \ .$$

LPs can harvest their `claimable_fees` at any time, at which point all collected fees will be transferred to the position owner and π reset to zero.

5.3.2. Deposits

Whenever an LP deposits liquidity (collateral and/or contracts) into a tick range, the `liquidity_net_change` ($\Delta\Phi$) of the position's lower and upper tick has to be updated to account for the newly deposited liquidity. Given an LP's net change $\Delta\phi(R) > 0$ in additional provided per tick liquidity the update equations are

$$\begin{aligned} \Delta\Phi(i_l) &\leftarrow \Delta\Phi(i_l) + (2w-1)\Delta\phi(R) \\ \Delta\Phi(i_u) &\leftarrow \Delta\Phi(i_u) - (2w-1)\Delta\phi(R) \end{aligned} \ .$$

where $w = 0$ or $w = 1$ when R is a bid- or ask-side order. Note that if either of the range order's ticks are not included in the active set of ticks, the new tick(s) are inserted into the linked list. Figure 3 exemplifies the net change in per tick liquidity given three range orders to the right-side of the market price.

5.3.3. Withdrawals

LPs can partially or fully withdraw their liquidity from their open positions at any time. Execution of a withdrawal effects the global per tick liquidity Φ by the liquidity withdrawn if the market price is within the order's range $[\tau_l, \tau_u)$. The following equation describes the required change during a withdrawal

$$\Phi \leftarrow \Phi - \alpha\, \phi(R)\, \mathbf{1}_{p \in [\tau_{i_l}, \tau_{i_u})}$$

where $\alpha \in (0, 1]$ is the percentage of the order's liquidity withdrawn. Moreover, dependent upon the current active tick the net change in per tick liquidity has to be updated. The adjustment in net change per tick liquidity is

$$\begin{aligned} \Delta\Phi(i_l) &\leftarrow \Delta\Phi(i_l) - \text{sign}(i_l, p)\, \phi(R) \\ \Delta\Phi(i_u) &\leftarrow \Delta\Phi(i_u) + \text{sign}(i_u, p)\, \phi(R) \end{aligned} \ ,$$

where the function sign : $[N] \times [0, 1] \rightarrow \{-1, 1\}$ is $+1$ if the price is less than the tick and -1 i.e.

$$\text{sign}(i, p) = \begin{cases} 1 & \text{if } p < \tau_i \\ -1 & \text{if } p \geq \tau_i \end{cases} \ .$$

5.4. Option Settlement

Since options on Premia v3 are European in nature, users can only exercise/settle option positions post-expiration. Options can be automatically settled after expiration by any external agent or software. An agent that settles an option or LP position for another user will be given a small fee for the action to compensate for payment of the EVM gas fees. This fee will be taken from the unlocked collateral in the option, meaning expired options with no value will have no incentive to be automatically settled.

LT Option Settlement When a long option is exercised, the option value is released to the option holder. When a short option is settled, the remaining `collateral`, net of option value, is released to the holder.

LP Position Settlement When an LP position is settled, any remaining `collateral`, `contracts`, and `claimable_fees` in the position are transferred to the position's owner, after paying any owed exercise value to the pool.

6. Margin

Premia v3 margin is a blend of attributes from traditional Reg-T [7] and Portfolio Margin [6] systems. A risk-based model is used to assess user positions in an isolated fashion. Lending markets are established exclusively for the purpose of providing capital to option underwriters, where each collateral type (eg. ETH, WBTC, USDC, DAI) has its own margin lending pool. Borrowers are in a first-loss position for exposure taken on, meaning the borrower's collateral is first used to cover any loss if a position becomes unprofitable. All profit is retained by the user borrowing capital, less capital usage fees.

6.1. Lender Capital

When depositing capital into a margin pool, each lender must select a deadline on which their capital is to be returned (`deadline`). A lender's capital can be borrowed at any time before the `deadline` (or indefinitely if no deadline is set). For example, if a lender's `deadline` is 30 days from now, this implies the lender's capital will be available for up to the next 30 days, and will only be used for options that expire between the current time and the `deadline`.

Any lender capital utilised in an option position is locked for up to the expiration of the position. Upon borrowing, a borrower pays an upfront commitment fee based on the utilisation of the total available capital in the pool, up to the option's expiration. When a borrower successfully closes a position, each lender's capital is unlocked and immediately made available for further lending or withdrawal. At any time, lenders may withdraw any of their capital that is not being utilised in the pool.

A lender's capital may be utilised for multiple options, so long as each expiration date is prior to the lender's `deadline`. All lenders for a specific expiration share in lending fees pro-rata. Additionally, lenders split principal risk of liquidated option positions, if and only if the Reserve Fund (discussed below) cannot fully cover losses. When a lender's `deadline` is passed, their capital will be reserved for withdrawal and will no longer be available to be borrowed for margin.

Early Withdrawals Lender's can withdraw their locked (utilised) capital at any time, provided there is sufficient available capital in the pool. In this case, the withdrawing lender must pay a commitment fee to borrow the capital from the other lenders in the pool, in order to unlock their locked capital.

6.2. Margin Positions

Margin borrowers pay a commitment fee to lenders, which is a simple annual interest rate for the amount of collateral borrowed, paid upfront at the time each position is opened. If a position is closed before expiration, unused interest charges will be refunded in the form of a rebate. The interest rate r is derived from the ratio of currently available reserve funds relative to the global exposure of borrowed positions. The interest rates are bounded based on r_{base} and r_{risk}.

The minimum interest rate, r_{base}, is established using an on-chain oracle for international risk-free money market rates. The maximum interest rate adds a risk premium, r_{risk}, to the minimum rate, based on the maximum expected risk incurred by lending capital in the pool. For each deadline, the interest rate can be calculated as

$$w = \frac{\tanh\left(a \cdot \frac{px - S_{fund}}{S_{fund}}\right) + 1}{2}$$

$$r = r_{base} + r_{fee} + w \cdot r_{risk}$$

where x is the amount to be borrowed, S_{fund} is the available reserve fund capital, $p \in [0, 1] = 1$ is the expected maximum risk probability, $a \in \mathbb{R}^{+} = 16$ is a steepness coefficient, and $r_{fee} = 0.5$ is the protocol fee. The parameters p, a, and r_{fee} are determined by governance procedures.

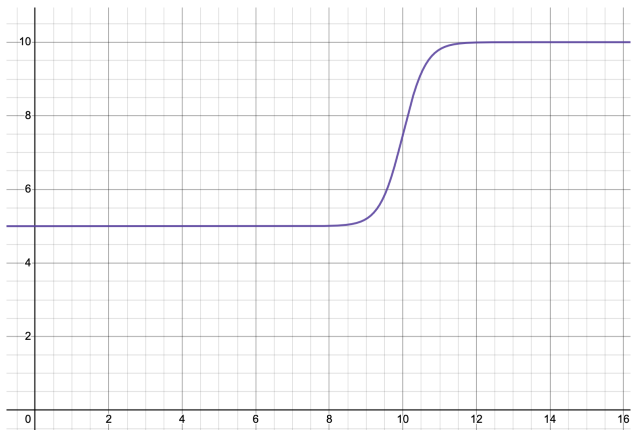

Figure 4: Interest Rate Curve: $S_{fund} = 10.0, r_{base} = 4.5, r_{risk} = 5.$[3]

Early Close Borrowers can close positions early, in exchange for any unrealized positive returns and a rebate position. The rebate position will track the amount of commitment fees earned by the liquidity returned to the pool, up until the `deadline` of the position. The user can withdraw their earned commitment rebates at any time from the pool.

Minimum Margin Minimum Margin (M_*) is a threshold value, dynamic over the lifetime of an option, which determines a position' s point of liquidation. The Minimum Margin for underwriting an option is calculated using a time-scaled, log-normal, 5% value-at-risk (VaR) for a single-tail that admits the option market's current implied volatility as an input value. VaR is a statistic that quantifies a potential loss with a given probability over a set time period.

All else being equal, as the number of days to expiration (DTE), IV, and/or moneyness of an option increases (de-

[3] A demonstration is provided in Desmos: https://www.desmos.com/calculator/iufb9bdch7

creases), the Minimum Margin requirement increases (decreases). A floor value for Minimum Margin is currently set to 3% of the underlying market price (strike price) for calls (puts), to limit leverage on far out-of-the-money options.

Value-at-Risk In order to compute the value-at-risk, the underlying process, S_t, is assumed to follow a geometric Brownian motion with zero drift and (implied) volatility, σ. The VaR for a z-score $z_{1-\alpha}$ with the level of significance $\alpha = 0.05$ can then be computed as

$$\text{VaR}_\alpha(p) = \begin{cases} \left(S_t\, e^{(z_{1-\alpha}\sigma\sqrt{\tau})} - K\right)^+ - p & \text{if call} \\ \left(K - S_t\, e^{(-z_{1-\alpha}\sigma\sqrt{\tau})}\right)^+ - p & \text{if put} \end{cases}$$

which represents the worst-case loss to the holder if the position was held until expiration. The VaR calculation is normalised for calls, since calls use the underlying as collateral. This is not required for puts since they use the numéraire as collateral. This VaR calculation becomes the basis for Minimum Margin

$$M_*(K,T) = \begin{cases} \max\left(c, \frac{\text{VaR}_\alpha(p)}{S_t}\right) & \text{if call} \\ \max\left(cK, \text{VaR}_\alpha(p)\right) & \text{if put} \end{cases}$$

where $c = 0.03$ is the floor scaling parameter. From the minimum margin calculation we are able to directly compute the Initial Margin Requirement (M_0) as

$$M_0(K,T) = \begin{cases} \max\left(rM_*(K,T),\ 1\right) & \text{if call} \\ \max\left(rM_*(K,T),\ K\right) & \text{if put} \end{cases}$$

where $r = 1.5$ is the initial margin multiplier.

Initial Margin The Initial Margin Requirement (M_0) is the capital required to be provided by the borrower when opening a sell-side position on margin and is simply 150% of the position's Minimum Margin at the time the position is created. If the Initial Margin Requirement at the time of opening a position is greater than the capital requirement for the fully collateralised position, margin is not available for the selected option. Initial Margin (ν_0) is the amount of collateral actually provided by a borrower when opening a position on margin. A borrower can provide more collateral than is required ($\nu_0 > M_0$), to decrease their interest fees and any potential chance of liquidation.

Collateral Value The Collateral Value (ν) and Minimum Margin for a position are both derived from the current market price of the option. A position can be liquidated if ν drops below M_* for the borrowed option ($\nu < M_*$). The current Collateral Value of a position is defined as

$$\nu = \nu_0 + \text{P\&L}^* - \zeta(x)$$

where P&L* is the full position's current unrealised profit or loss.

Note: Minimum Margin is not a static value after a position is created, rather, it updates according to the market conditions. This means a margin borrower should be aware of not only their current Collateral Value, but also of an option's Minimum Margin value to avoid liquidation. At the discretion of borrowers, capital can be manually added or automated software can be used to provide additional capital to at-risk positions to prevent liquidation.

6.3. Liquidations

Liquidations do not occur automatically on-chain. For this reason, positions must be liquidated by a third party. Anyone can liquidate an at-risk position on-chain and collect a small fee for doing so. Liquidators do not take ownership of a liquidated position, rather, they are rewarded with a fee derived from the size of the position, since they are required to cover the EVM transaction fee. Automated software is deployed alongside PREMIA V3 to monitor and liquidate user positions, but the code will also be open sourced for third party liquidators to participate and compete for fees.

Upon liquidation, ownership of the position is transferred from the borrower to the Reserve Fund. Any user collateral remaining in the position is forfeited as a liquidation fee. This liquidation fee is used to mitigate the risk of the margin system holding the position until expiration, as liquidated positions are *not* closed pre-expiration. This has the added benefit of reducing price manipulation and cascading liquidations, which can disrupt market price with negative feedback loops. Borrowers also have the option to add margin to any open position to prevent liquidation.

If a lender's capital has been utilised on a position that has been liquidated, it has effectively been time locked until the option's expiration. Since all positions are fully-collateralised at the *exchange level*, there is never a risk of insolvency for the option.

6.4. Reserve Fund

Collateral in the Reserve Fund is meant to absorb the profit-or-loss of liquidated positions. Since lenders only provide capital to option positions that have times to maturity less than their `deadline`, the margin system is able to settle positions and abstract profit-or-loss variance from lenders before they are able to request a withdraw. Simply stated, the Reserve Fund is in a first-loss position against lenders' principle. Additionally, it is capable of distributing excess reserves as supplementary yield to lenders, akin to dividend payouts. If the Reserve Fund were to ever be insolvent, lenders' principle could be exposed pro-rata to loss on liquidated positions.

Reserve Fund Accounting Funds in the reserve are utilised in order of option expiration. For options that have the same expiration, options are settled on a first-come, first-settle basis. Lenders will continue to accrue interest fees on liquidated positions for the entire time their capital is utilised, paid by the Reserve Fund at the time upon settlement. The Reserve Fund's liquidity state is then updated as

$$s \leftarrow (s + \text{P\&L})^{+}$$

where P&L is the option's realised profit-and-loss at expiration (net of all fees collected and paid).

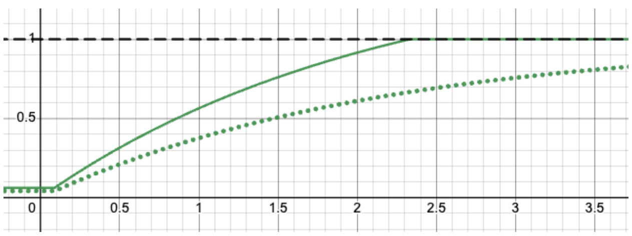

Figure 5: Minimum and Initial Margin for ATM Call, $t = 30$.

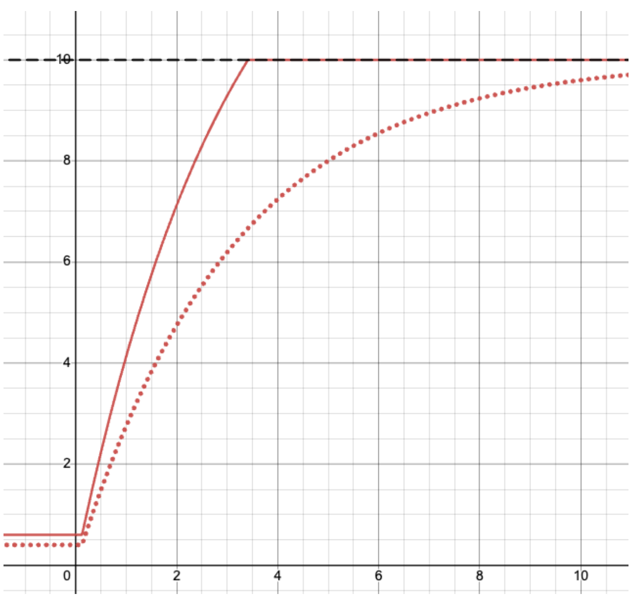

Figure 6: Minimum and Initial Margin for ATM Put, $K = 10$, $t = 30$.[4]

[4]A demonstration is provided for calls and puts in Desmos: `https://www.desmos.com/calculator/d4xs4s2vbh` and puts `https://www.desmos.com/calculator/hxlf0d4z8g`.

7. OTC Liquidity

Market-makers on-chain have to optimise between active and passive liquidity management, factoring EVM transaction fees into their calculations. The over-the-counter (OTC) quote system enables vaults and market-makers on- or off-chain to provide quotes to users or aggregators, which can then be fulfilled through the exchange alongside range orders. This increases liquidity across strikes and maturities and enables professional market makers to optimise for both execution price and transaction fees. However, these market makers will need to source their own trading volume as off-chain quotes will not yet be displayed to users directly interacting with a pool's smart contract.

Volatility-Based Pricing Institutional traders often price options in terms of volatility or implied volatility. Quote-based liquidity enables professional traders to provide volatility-based quotes to takers, without paying the EVM gas fees required to manage on-chain liquidity positions.

8. Vaults

The complexity of pricing and strategy creation is a major challenge within option markets, which often drives away passive users. Vaults enable the most passive of users to automatically participate in options market making and yield capture strategies created by third parties. Vaults can utilise standard range orders or the quote-based liquidity system to execute strategies on behalf of users, splitting returns across liquidity within the vault.

The OTC quote system enables specific vaults to provide liquidity across the entire volatility surface, similar to PREMIA V2, further decreasing liquidity fragmentation and increasing capital efficiency. Vaults can additionally be created by any third-party provider, enabling anyone to participate in strategy discovery and monetisation.

References

[1] H. Adams, N. Zinsmeister, M. Salem, R. Keefer, and D. Robinson. Uniswap v3 core. *Uniswap, Tech. Rep.*, 2021. URL `https://uniswap.org/whitepaper-v3.pdf`.

[2] G. Angeris and T. Chitra. Improved price oracles. In *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*. ACM, oct 2020. doi: 10.1145/3419614.3423251. URL `https://doi.org/10.1145%2F3419614.3423251`.

[3] S. Dawson, N. Forster, D. Romanowski, and M. Spain. Lyra, 2021. URL `https://www.lyra.finance/files/whitepaper.pdf`.

[4] M. Egorov and C. F. S. S. GmbH). Automatic market-making with dynamic peg. June 2021.

[5] Enso, Froggie, Htmh, Karlis, and Wolfy. Premia amm. *Premia Finance*, 2021. URL `https://premia.finance/amm.pdf`.

[6] A. Hayes. Portfolio Margin. July 2022. URL `https://www.investopedia.com/terms/p/portfolio-margin.asp`.

[7] W. Kenton. Regulation T (Reg T). November 2020. URL `https://www.investopedia.com/terms/r/regulationt.asp`.

[8] G. Liao and D. Robinson. The dominance of uniswap v3 liquidity. *Uniswap*, 2022. URL `https://uniswap.org/TheDominanceofUniswapv3Liquidity.pdf`.