LendBook
a Lending Limit Order Book

Breaking Lab *

January 25, 2024

Abstract

A lending limit order book is a non-custodial and permissionless lending protocol that enables users to borrow assets from limit orders collateralized by their own limit orders. This new financial primitive offers users multiple benefits: stop loss orders with guaranteed stop price, low liquidation penalty, high loan-to-value and leverage and interest-bearing limit orders. The protocol is immune to the risk of bad debt and can operate without the need for off-chain governance.

Current issues with lending protocols

Lending protocols provide users with the ability to lend and borrow cryptoassets in a decentralized, permissionless, and trustless manner. However, their growth has been impeded by a common birth flaw: the risk of incurring bad debt. A bad debt appears when the price of the deposited collateral rapidly falls and its value no longer covers the outstanding loan, putting the protocol at risk of insolvency

*contact: pr3vert. We thank Mathis Gontier Delaunay, Hamza El Khalloufi and Kirk Hutchison for useful discussions and comments.

and bank run. The loss can be exacerbated by the lack of sufficient liquidity in the pools in which the collateral assets are exchanged to cover borrowed assets during liquidation events.[1]

To mitigate this existential risk, lending protocols impose various constraints on the borrowers side, such as high collateral-to-debt ratios and high liquidation costs. They also limit lending markets to high-quality assets. These constraints impair users' experience by making borrowing more expensive and riskier and by restricting the range of borrowable assets. Despite many innovative features introduced in the sector since its emergence in 2018, significant improvement of users experience is still awaiting a foolproof solution for the risk of bad debt.

LendBook is a new lending primitive which eliminates the insolvency risk, can achieve full decentralization by getting rid of offchain risk management and brings along the way a host of new benefits for lenders and borrowers.

1 What is a lending limit order book

A lending limit order book (LLOB) is a non-custodial, peer to peer and permissionless order book that enables users to post limit orders and borrow orders' assets. Borrowing positions are collateralized by limit orders posted on the other side of the order book: borrowed assets from buy orders are collateralized by the assets deposited in sell orders, and reciprocally.

Figure 1 presents the double queue organization of a typical central limit order book. On the left-hand side, makers place buy orders which traders find profitable to take if the price decreases below their limit prices. On the right-hand side, they place sell orders which are filled if the price increases above their limit prices. Rectangles' height indicate how much makers have committed in their limit orders.

Fig. 2 shows how a lending primitive is attached to a limit order book.

[1]See RiskDAO for examples of bad debt events.

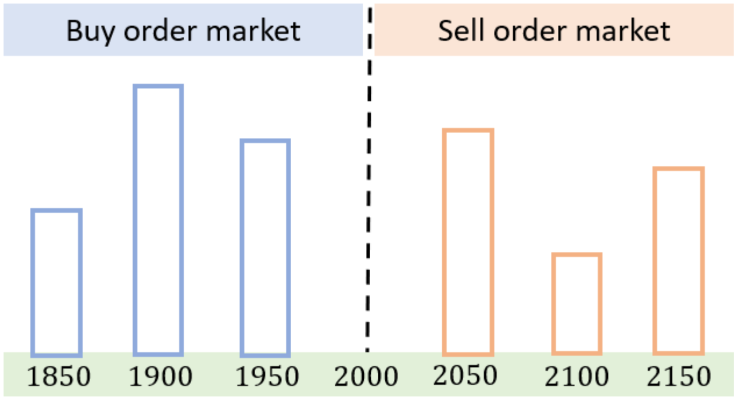

Figure 1: A graphical representation of a central limit order book (market price is 2000).

Makers allow traders to borrow their assets deposited in the book in exchange of an interest rate. The rectangles with a blue background represent the assets borrowed from the orders at the same limit price.

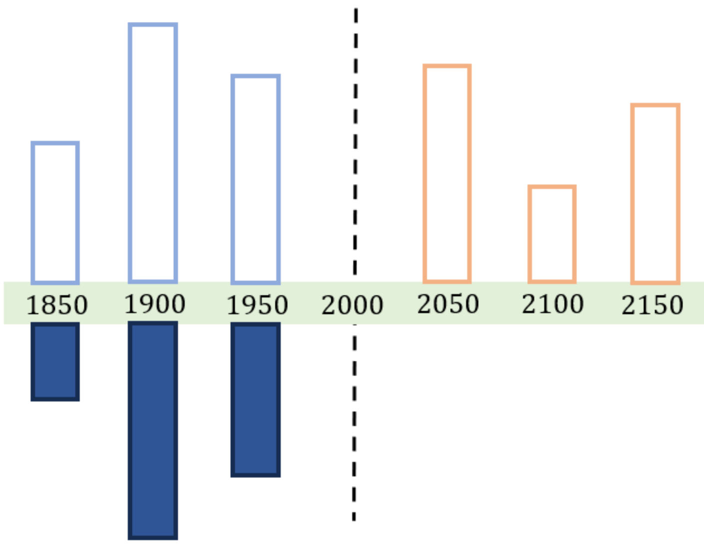

Figure 2: A central limit order book with lending functionalities. Rectangles with a blue background represent the assets borrowed from the buy orders at the same limit price.

The protocol ensures that any position borrowing from a pool of orders can be liquidated when part or all assets of the pool are taken. The alignment of the two events significantly streamlines the settlement process for all parties.

Example

Let's illustrate how a lending operation works in the simplest case of two actors and two orders. Market price is 2000. Alice posts a buy order of 3 ETH at price 1900 USDC. To do so, she deposits $3 \times 1900 = 5700$ USDC in the protocol's USDC pool. Bob is willing to borrow 3800 USDC from Alice's buy order. He places a sell order of 2 ETH at 2200 (or any price greater than market price) and deposits 2 ETH in the ETH pool. With 2 ETH of collateral, he can then borrow $2 \times 1900 = 3800$ USDC from Alice. The financial flows are summarized in Fig. 3.

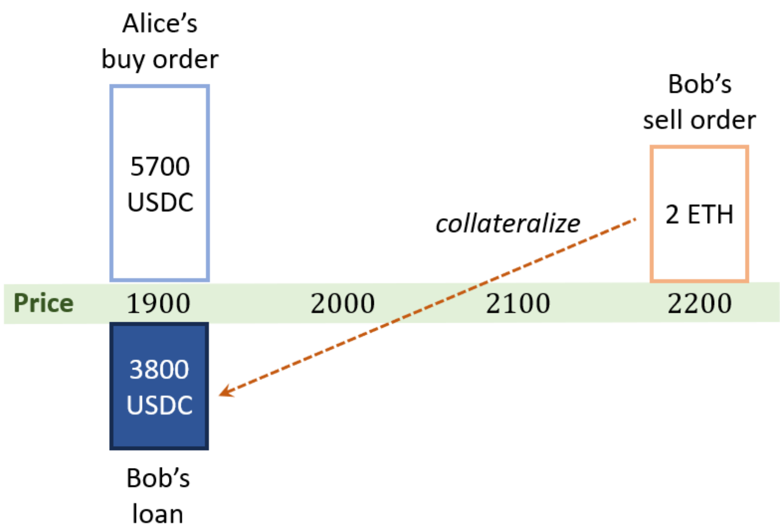

Figure 3: Bob borrows 3800 USDC from Alice, collateralized by his sell order at 2200.

If the price decreases to 1900, a taker swaps Alice's remaining 1900 USDC for 1 ETH, which triggers the closing of Bob's position. Bob keeps the borrowed USDC and Alice is given 2 ETH taken from Bob's collateral (see Fig. 4).

Nothing changes for Alice compared to a non-borrowed buy order. Importantly, Bob's assets are not market sold but directly transferred to Alice who is paid back with the collateral.

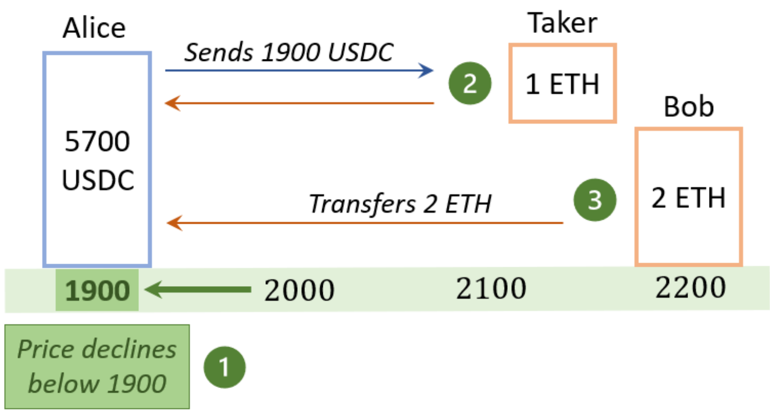

Figure 4: If the price decreases to 1900 (1), Alice's remaining assets are taken (2). Bob's collateral is transferred to Alice to complete the filling (3).

Before reviewing the benefits of implementing a lending order book, let's dive into the details of its functioning.

2 Functioning

To concentrate liquidity, makers are permitted to deposit assets within a restricted range of limit prices. A pool of limit orders groups all assets deposited at the same limit price allowing users to deposit and borrow from the pools, rather than from individual orders. Admissible limit prices are connected through a multiplicative step, the size of which varies based on the asset's nature and volatility. For instance, a 15% step is used for highly volatile and long-tail assets, 10% for volatile top-tier assets, and a range of 5 to 1% for correlated and pegged assets. The aim is to restrict the number of borrowable pools to between 3 and 5.

Illustration with a step of 10%: Market price is 2000. Alice may deposit assets in buy orders at limit price 1900, $1900/1.1 = 1727$ or $1900/1.1^2 = 1579$, and so on, but not at limit prices in between.

Furthermore, order books will trade an asset pair in which only one of the two assets can be borrowed and the other serve as collateral. In other words, assets which serve as collateral cannot be borrowed and borrowable assets cannot serve as collateral.

Example: In the ETH/USDC market, Bob can borrow USDC deposited by Alice in buy orders but Alice cannot borrow ETH deposited by Bob in a sell order.

The operations are governed by four additional rules.

R1. When part or all assets in a pool of orders are taken, positions borrowing from the pool are closed out.

In previous example, Bob's position is closed out when Alice's buy order is taken at 1900.

R2. Liquidated borrowers pay a small liquidation fee to lenders (1 or 2%).

The liquidation fee and the interest rate (see infra) are the two sources of income paid by borrowers for the liquidity service offered by lenders. Liquidation fees compensate lenders for receiving the collateral.

Example continued with a 1% liquidation fee: Alice's buy order is filled for 3 ETH. Bob's collateral transferred to Alice is $2 \times 1.01 = 2.02$ *ETH.*

R3. If borrowers' limit orders, which assets serve as collateral, are filled, their borrowing positions are closed out.

The closing of the borrowing position guarantees that the type of assets serving as collateral always matches the type needed in case of liquidation.

Example (variant): Bob's sell order is filled first for $2 \times 2200 = 4400$ *USDC, of which 3800 are kept by the protocol to close his borrowing position from Alice's buy order. Bob is left with 3800 (borrowed assets) 600* $= 4400$ *USDC.*

R4. Orders which assets are taken are automatically replaced on the opposite side of the order book.

The converted assets are replaced at a limit price specified by the maker, or, in the absence of such specification, at a default limit price. This default is set at 10% above or below the previous limit price, depending on the nature of the order. Lenders can then change the limit price if they wish or withdraw the assets.

Example (follow up): Alice's buy order is filled for 3 ETH. The protocol relocates the ETH in a sell order which limit price is by default is 2090 (1900 + 10%).

3 Interest rate model

3.1 Interest rate curve

One interest rates exists for every active pool of borrowable assets.

Example (continued): Market price is 2000. Three active pools exist with buy orders: at limit prices 1900, 1750 and 1620. For each pool, an interest rate is quoted.

It is algorithmically adjusted to the real-time supply and demand of assets. The market's utilization rate (U) of pool i is the ratio of the total borrowed assets B_i to the total supplied assets D_i:

$$\mathrm{U}_i = \frac{\mathrm{B}_i}{\mathrm{D}_i}$$

There exists as many utilization rate and interest rate than active pools. The higher the pool's utilization rate, the higher the interest rate R_i. The interest rate curve is split in two parts around an optimal utilization rate U^*. Before the slope is small, after it begins rising sharply:

$$U_i \leq U^* : R_i = \alpha + \beta \frac{U_i}{U^*}$$
$$U_i > U^* : R_i = \alpha + \beta + \gamma \frac{U_i - U^*}{1 - U^*}$$

with α, β and $\gamma > \beta$ three positive parameters. R_i is paid by users borrowing from pool i. Lenders earn an interest rate scaled down by the pool's utilization rate:

$$R_i^l = U_i R_i$$

3.2 Interest-based liquidation

As time goes on, the borrowed amount increases due to the compound interest rate. A collateralized position can become under-collateralized and subject to liquidation. This type of liquidation is distinct from the price-based liquidation assumed by makers and explained supra.

Borrowers' positions are eligible for liquidation when their excess collateral (EC) is zero or negative. User i's EC is the difference between the value of assets deposited as collateral (CV) and the required collateral (RC) multiplied by the minimum collateralization ratio (CR):

$$\mathrm{EC}_i = \sum_i \mathrm{CV}_i - \mathrm{CR} \times \sum_i \mathrm{RC}_i > 0$$

The minimum collateralization ratio is the lowest allowable ratio of the value of collateral a borrower must deposit to take out a loan. This is a safety mechanism to protect lenders from the borrower's insolvency due to a growing debt burden. Since the interest rate is slowly increasing, CF can be set close to 100%.

Example with CF set to 101%: Alice deposits 3800 USDC in a buy order at 1900. Bob deposits 1.1 ETH in a sell order and borrows 1900 USDC from Alice. Excess collateral is positive and equal to 1.1 - 1.01 × 1 = 0.09 ETH. After six months, his loan plus interest rate climbs to 2071 USDC, which requires a collateral value of 2071/1900 = 1.09 ETH. Bob's excess collateral becomes 1.1 - 1.01 × 1.09 = 0 ETH. His loan is eligible for liquidation.

The liquidation is at the initiative of external actors who receive in exchange up to a 5% bonus.

Example (continued): Price is 2000. A liquidator pays back Bob's debt equal to 2071 USDC. He receives in exchange 1.05 × 2071/2000 = 1.087 ETH. Bob is left with 1900 USDC and 1.1 - 1.087 = 0.013 ETH.

4 Benefits of using LendBook

The benefits of a lending order book are multiple: low liquidation penalty, high leverage, stop loss orders with guaranteed stop price, programmability of leverage strategies, automated market making and minimized governance. Let's review them one by one.

4.1 Low liquidation cost

The prevalent liquidation mechanism in lending protocols allows a liquidator to repay a fraction of the borrower's debt and acquire its collateral at a discount. In LendBook, when the price crosses a limit price, the closing of a borrowing position does not rely on the active monitoring of liquidators but on that of takers. Traders, by taking the non-borrowed part of the assets, initiate the internal transfer from the borrowers to the lenders. Since lenders agree to receive the collateral as a payment, the protocol does not need to incentivize bots to liquidate unhealthy positions on time. The borrower only pays small liquidation fee rate (1 or 2%), with the goal to compensate lenders for receiving the collateral and give borrowers incentives to repay their loan before liquidation.

4.2 High Loan-To-Value

The maximum Loan to Value (LTV) ratio defines the maximum amount of assets that can be borrowed with a specific collateral. For example, a maximum LTV of 80% in the ETH/USDC market means that users can borrow at most 0.80 USDC for every 1 USDC worth of ETH deposited as collateral.

As the price at which borrowers are liquidated is fixed (it is the limit price of the pool of orders from which they borrow), the maximum LTV is:

$$\text{max LTV} = \text{LLTV} \times \frac{\text{pool's limit price}}{\text{current price}} \tag{1}$$

with LLTV the limit LTV that users can approach by borrowing from a pool which limit price is arbitrarily close to market price. To prevent early liquidation, the LLTV is set below the inverse of the minimum collateral ratio below which positions are liquidated: $\text{LLTV} < 1/\text{CR}$. With $1/\text{CR} = 1/1.01 = 0.99$, the LLTV is set to 0.98.

Example (continued): With 1 ETH deposited in a sell order, Bob can borrow from Alice's buy order at 1900 up to LLTV × 1900 USDC. His maximum LTV is:

$$\text{LLTV} \times \frac{1900}{p}$$

The closer the ETH price p to the limit price, the closer the max LTV to LLTV. For LLTV = 0.98, max LTV is 0.93 for p = 2000 and 0.97 for p = 1920.

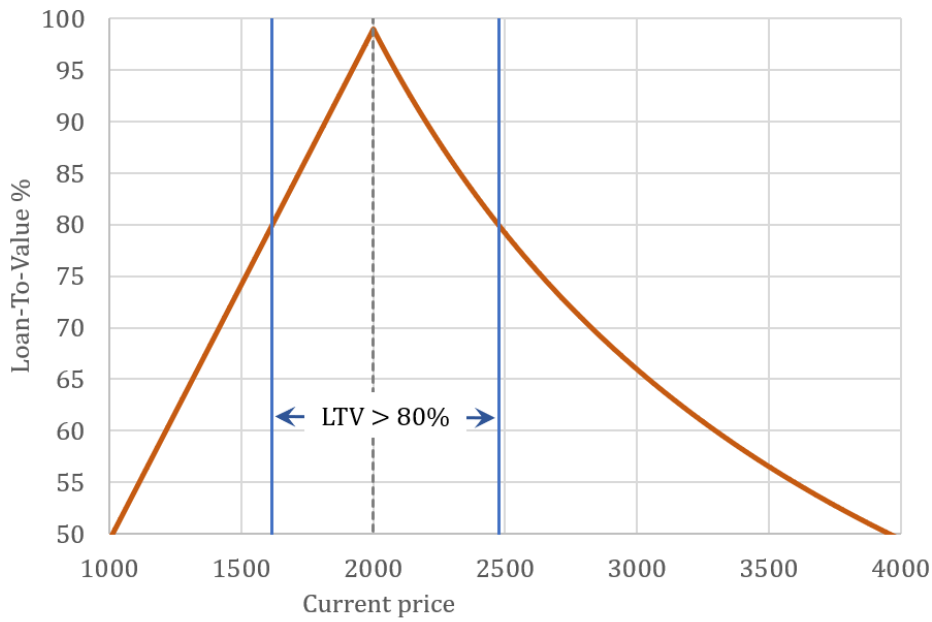

Figure 5: maximum LTV in function of the distance of the pool's limit price to current price (2000). The LTV for limit prices below the market price are valid for the X/Y market in which Y is borrowable and the the LTV for limit prices above the market price for the Y/X market for which X is borrowable.

Fig. 5 shows the maximum LTV in function of the distance of the pool's limit price to current price (set to 2000). LTV can approach 98% for limit prices close to current price. For reference, the price interval [1615, 2475] indicates the limit prices at which assets can be borrowed with a max LTV higher than 80% (the max LTV of the ETH market in Aave V3).

4.3 High Leverage

LendBook enables leverage factors a magnitude higher than what other lending protocols offer. To understand how, let's examine how traders leverage their position. As in other protocols, they can borrow Y and swap them to amplify their position in X, or they can borrow X and swap them to short X. Borrowers can easily do loops of borrowing and swapping to amplify their leverage.

High LTV translates into high leverage. Abstracting from gas and swap costs, the n-loop leverage factor is:

$$1 + \text{LTV} + \text{LTV}^2 + ... + \text{LTV}^n$$

Assuming borrowers could infinitely loop at the same limit price, their maximum leverage would be:

$$\frac{1}{1 - \text{max LTV}}$$

As the limit price gets closer to the market price, the max LTV tends to the limit LTV 0.98 and the theoretical maximum leverage to 50.

Fig. 6 shows the maximum leverage factor as a function of the distance of the limit price to the market price. Buy orders, which limit prices are lower than market price, give traders a way to leverage-long the base token. The maximum leverage is 50 for a buy order limit price of 2000. Borrowers can still achieve a significant leverage for limit orders far away from current price. The maximum leverage is 10× for limit prices 1835 and 2180 and 5× for limit prices 1635 and 2450.

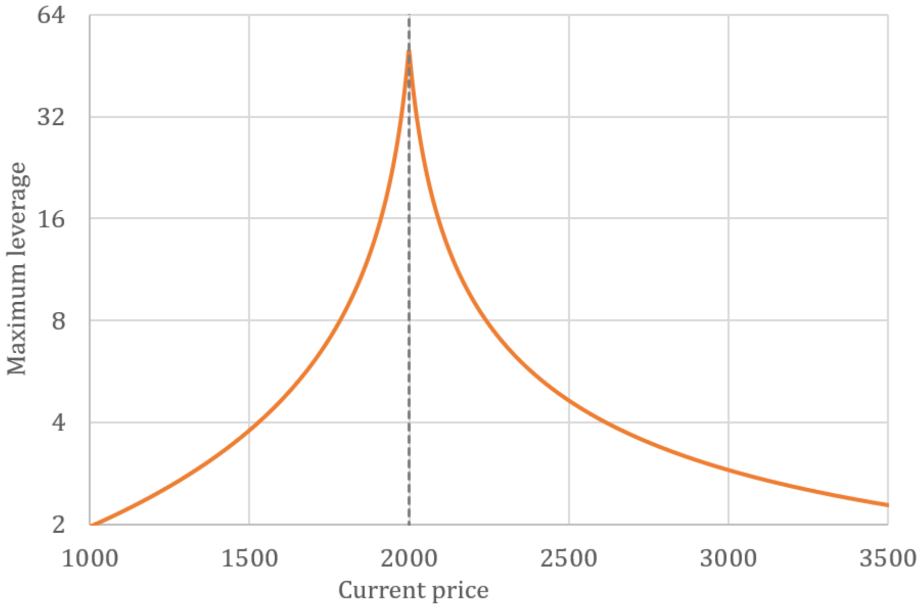

Figure 6: Distance of limit price to market price (2000) and maximum leverage (base-2 log scale). The leverage factor for limit prices below the market price are valid for the X/Y market in which Y is borrowable and the the leverage factor for limit prices above the market price for the Y/X market for which X is borrowable.

4.4 Stop loss and take profit orders

4.4.1 Stop loss

A stop-loss order allows traders to close long positions by selling the assets or a short position by buying the assets. In LendBook, users open stop-loss orders by borrowing assets from limit orders. The stop price in case of price decrease (or increase) is the limit price of the buy (sell) order from which they borrow.

Example: Bob deposits 2 ETH in a sell order and borrows from Alice's buy order at 1900: $0.98 \times 2 \times 1900 = 3724$ *USDC. if the price decreases to 1900, Alice's buy order is taken. Bob keeps the 3724 USDC and gives up his 2 ETH. This is as if he benefits from a stop loss (sell ETH when its price decreases) at the guaranteed price of 1900. His stop price is Alice's limit price.*

In traditional or crypto finance, once the stop price is met, the stop loss order becomes a market order and is executed at the next available price. The obtained price can be significantly less favorable than the specified price when markets move fast. Here the stop price is guaranteed by the filling of the sell order at the limit price.

4.4.2 Take profit

In addition, by posting their collateral in the order book, borrowers can program in advance the price at which exit their strategy, which is the limit price of their collateral order.

Example: By depositing ETH in a sell order at 2200, Bob benefits from a take-profit at the same price. If the market price increases to 2200, Bob's sell order is taken first. His 2 ETH are exchanged against 4400 USDC from which 3724 are used to pay back his borrowing position (cf. Rule R3). He keeps the 3724 USDC borrowed from Alice and earns a profit of 676 USDC.

Setting in advance an exit price is an integral part of risk management in case of leveraged position.

Example: Bob borrows 3724 USDC from Alice, exchanges the amount for 1.86 ETH at market price 2000 and deposits the amount in his sell order at 2200. He then borrows $0.98 \times 1.86 \times 1900 = 3445$*, sells for 1.72 ETH and deposits the proceeds in his sell order. If the price hits 2200, his sell order is filled and his borrowing position is closed out. The protocol pays back Bob's debt of* $3724 + 3445 = 7169$ *USDC with the* $2200 \times (2 + 1.86 + 1.72)$ *ETH = 12276 USDC of his sell order. Bob's leveraged profit is* $12276 - 7169 - 2 \times 2200 = 707$ *USDC.*

Fig. 7 shows the price interval over which Bob makes a profit or a loss and at which prices his position is closed out in both directions.

The combination of an order book with a lending protocol unlocks a rich set of strategies that borrowers can fine-tune and program in advance. They can spread their position over several collateral orders and pools with different limit prices. This way, their borrowing can be progressively reduced as the price

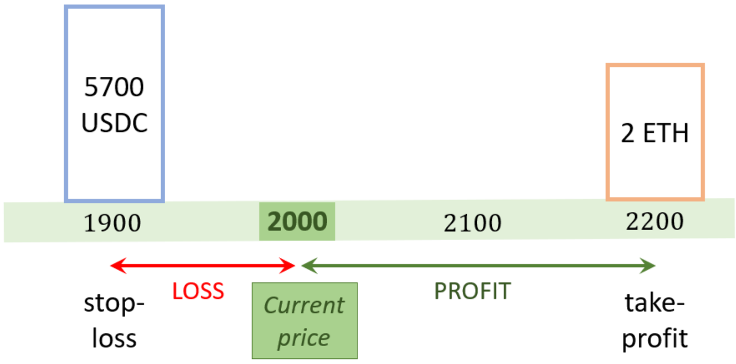

Figure 7: When Bob borrows from Alice at 1900 and places a collateral limit order at 2200, his leverage is closed out for a profit if his limit order is taken first and for a loss if Alice's buy order is taken first.

reaches well-specified thresholds.

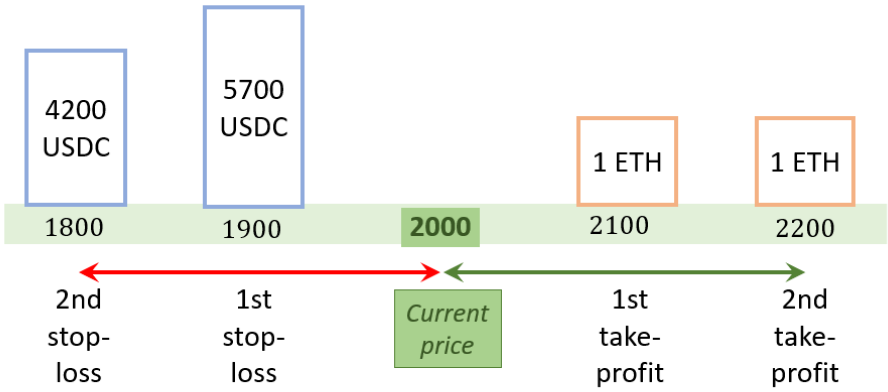

Figure 8: Bob borrows USDC from a pool et 1900 and another one at 1800, collateralized by a sell order at 2100 and another one at 2200.

In Fig. 8, Bob is gradually liquidated at prices 1900 and 1800. If the price increases, he progressively takes profit at 2100 and 2200.

Another strategy for borrowers consists in reposting their borrowed amounts on the order book to gradually raise their leverage as the price moves.

In Fig. 9, Bob places a sell order of 2 ETH at 2200, borrows 3500 USDC from Alice at 1800 and reposts the amount in a buy order at 1900. If the price decreases to 1900, Bob's USDC are taken in exchange of $3500/1900 = 1.84$ ETH.

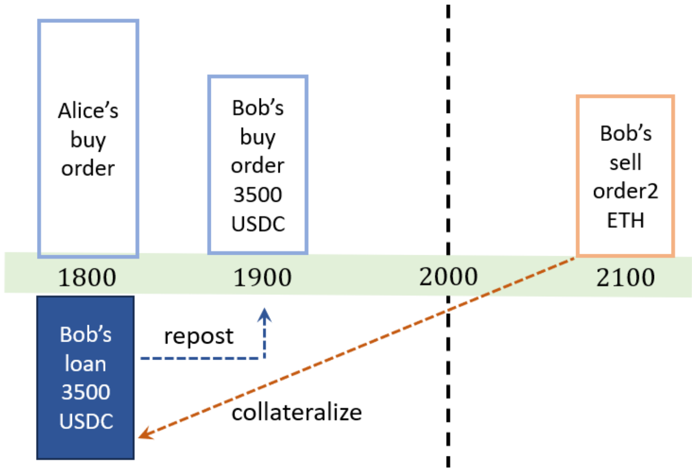

Figure 9: Bob replaces his borrowed USDC on the order book to leverage long his position if the price declines to 1900.

Bob is now leveraged long ETH. If the price reverts and increases above 1900, Bob profits.

4.5 Automated market making

Orders which assets are filled are automatically replaced in the order book (Rule 4). Lenders choose in which pool and at which limit price their assets are replaced. By default, their liquidity is replaced in the nearer pool on the other side of order book.

Example with pools spaced with a 10% step: Market price is 2000. Alice deposits 5700 in a buy order which limit price is 1900. Once filled, the protocol relocates the 3 ETH in the nearer pool of sell orders which limit price is 2090.

This feature allows makers (or protocols built on LendBook) to program in advance at which price they are willing to sell back the assets after a buy or buy back them after a sell. The default strategy with automatic replacing in the nearer opposite pool is similar to what liquidity providers do in Uniswap-type Automated Market Makers (AMMs), except that instead of earning a fee rate, makers earn the spread on top of the interest rate paid by borrowers.

Example (continued): After hitting 1900, the price reverts and crosses 2090. Alice's sell order is taken for 3 $\times$ 2090 = 6270 USDC. Her net profit is 6270 - 5700 = 570 USDC. Her profit rate is 570/5700 = 10%.

From there, two types of lending strategies are made possible. In the first type, lenders act as market makers and post limit orders closed to current price. They constantly replace their filled order on the other side of the market to earn the spread in addition to the lending return rate. In the second type, lenders follow single-sided Aave-style strategies. They earn a return for their deposited assets but avoid conversion by posting limit orders far from current price and/or by withdrawing their funds before their orders are filled.

4.6 Absence of bad debt

A major implication of borrowing assets from limit orders is the dramatic simplification and high safety of the liquidation process. Borrowing positions cannot go under-collateralized even in case of strong and rapid price action, gas fee spike, or blockchain congestion/downtime. While the risk of rapid price variation persists, it is borne by the maker of the limit order, creating an opportunity cost for them. Although this cost is inherent in all limit order books, in LendBook, makers are compensated through an interest rate and liquidation fees.

The fact that lending pools stay well collateralized under any market conditions brings many benefits at the UX and governance levels. In particular, there is no need for supervision by third party, funding a safety module, liquidation costs, or borrowing restrictions such as supply caps, borrowing caps and low loan-to-value.

4.7 On-chain and market-driven risk management

Another key advantage of a LLOB is that risk management is on-chain and market-driven instead of being off-chain and relying on centralized processes.

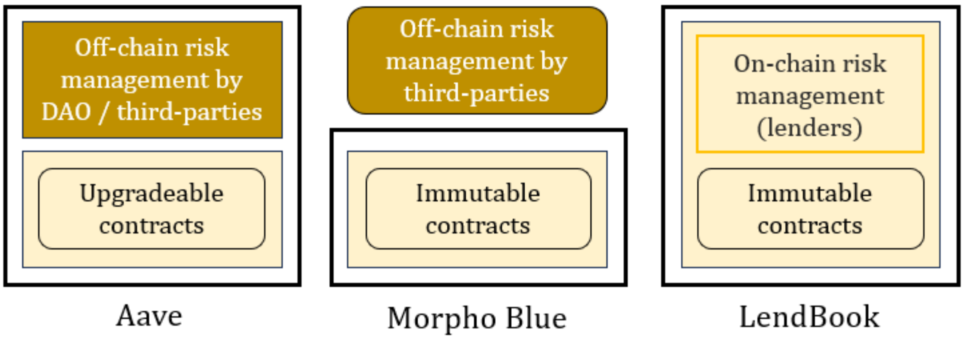

Figure 10: Comparison of lending protocol with regard to risk management.

Diagram 10 compares how two prominent lending protocols and LendBook manage the risk of bad debt. Whereas risk management is assumed by the DAO in Aave with the help of external agencies and is externalized in Morpho Blue, it is directly handled in LendBook.

To understand the implications, let us compare how high-volatility pools are managed in LendBook and other lending protocols. DAOs and risk experts typically set lower collateral ratios in high volatility pools to give liquidators enough time to close risky positions and prevent the creation of bad debt in case of price fall.

In LendBook, lenders handle the risk of volatile assets by migrating their capital to limit prices far enough from market price to reduce the risk of their assets being converted at an unfavorable rate. The migration of capital produces two effects. First liquidity is scarcer at limit prices closer to market price which drives the interest rate up and rewards lenders for the additional risk they take. Second, by borrowing in limit prices far from current price, borrowers benefit from lower LTV as in other lending protocols (see formula 1). The difference is that the lower LTV is not predetermined by an off-chain decision but the result of market forces.

Capital migration may also happen in case of increased volatility or bad news affecting the asset, information that other DAO-based lending protocols fail to integrate or only with long delays.

4.8 Oracleless lending

In V1, a price oracle is used to prevent users from taking buy orders which assets are partially borrowed at a loss.

Example: Market price is 2000. Alice deposits 4000 USDC in a buy order at 1900. Bob deposits 2 ETH in a sell order and borrows 3800 USDC from Alice's buy order. Alice's limit order cannot be filled as long as the price feed indicates a market price above 1900.

This restriction safeguards against premature liquidations by external parties. While filling orders at an unfair price is costly, the cost is mitigated by filling a small quantity, or even negated if the taker is the maker/lender. This could result to a profit at the expense of borrowers.

Example (continued): In absence of restriction, Alice could fill the remaining assets in her buy order at market price 2000, which would liquidate Bob's position. Bob is left with 3800 USDC in exchange of 2 ETH worth 4000 at current price. Alice's profit is (2 + 200/1900) × 2000 - 4000 = 210 USDC.

It is possible to guarantee that limit orders are executed at a profit without relying on an oracle. Consider a trader who is willing to take the available amount Y from a buy order which limit price is L_b. The filling must be preceded by the taker placing during a short time window the amount Y/L_b in a sell order at the same limit price L_b. If the sell order is unprofitable to take ($(p - L_b)Y/L_b < 0$), this means that the order offering Y at price L_b is profitable. In the converse case ($(p - L_b)Y/L_b > 0$), the sell order will be filed at the expense of the maker. The mechanism infers the market price and imposes a penalty on users trying to liquidate well collateralized loans.

Example (variant): To fill the remaining assets in the buy order at market price 2000, the taker has first to post a sell order with the amount 200/1900 = 0.11 ETH at limit price 1900. If the market price is lower than 1900, the sell order cannot be filled at a profit (0.11 × (p - 1900) < 0) and the taker will be authorized to fill the buy order. Conversely, if the market price is higher than 1900, the sell *order will be filled and the taker will lose her stake.*

Oracleless lending will be implemented in the V2 with the aim of addressing the market of long-tail assets.

4.9 Governance minimization

The governance activity of lending protocols has significantly grown and become more complex over time. Managing the risks of pools has been progressively delegated to experts, whose mission is to monitor the pools' risk and update their risk parameters. This involves assessing multiple risk factors, such as the on-chain liquidity of assets, price volatility, and market capitalization.

Governance-driven risk management hinders protocols from scaling horizontally. The more assets are listed, the greater the number of risk parameters that need to be monitored and updated in real-time. The DAO, with its limited scope for attention and complex decision process, becomes a bottleneck in expanding to more chains and assets.

Moreover, despite committing considerable resources and expertise to risk management, protocols' solvency is still at risk of a lack of due diligence or governance failure. Lending protocols have also implemented and funded sizeable financial buffers to absorb shortfall events and protect lenders from bad debt. Those safety measures mitigate solvency risk at the expense of token holders.

In contrast, the functioning of a LLOB is fully algorithmic and automated. As pools' solvency does not rely on team's interventions or governance by a DAO, full decentralization becomes a credible objective which LendBook will actively pursue. The protocol will ultimately be governance free with non-upgradeable smart contracts and parameters set at the time of contract deployment.

No governance process will be needed to whitelist approved tokens. Markets will be created permissionlessly by calling a factory contract. The number of assets that could be listed is only limited by the existence of a reliable price feed in the V1.[2] The V2 will expand to long-tail assets by getting rid of price oracles.

5 Conclusion

Lending protocols are an essential building block for blockchains' applications. They have grown to represent tens of billions of dollars in value. However, the vast majority of this value is held in smart contracts which management is still partially centralized. A complete decentralization process has failed so far, due to a persistent risk of insolvency, which management creates points of centralization.

LendBook's immunity to insolvency risk marks a significant advancement in the space. There is no concept of bad debt that might need to be absorbed by a DAO treasury / insurance fund or socialized across lenders. There is no trade-off in case of liquidation between the costs incurred by borrowers, liquidators' incentives and lenders' safety. The radically innovative design unlocks many new features like high LTV and leverage, borrowing programmability and interest-bearing limit orders. This also makes possible the protocol to operate in a fully decentralized way.

[2] Aave V3 currently lists 20 or so assets on Ethereum. The additional list of admissible tokens with a Chainlink feed includes SHIB, GRT, SAND, APE, CVX, ANKR, SUSHI, RDNT, BADGER and PERP.

Appendix A: Excess collateral and excess liquidity

Users' excess collateral

Excess collateral of a user is the sum of all base assets deposited in limit orders minus the assets required to fully collateralize her borrowing positions.

Excess collateral must always be positive for all users. Failing to be true, user's non-borrowed deposited assets could be insufficient to cover required transfers in case of liquidation of user's own positions.

More formally, suppose user i places sell orders $\{(x_s, p_s); s = 1, ..., S\}$ in S pools of sell orders and borrows from T pools the assets $\{(y_t, p_t); t = 1, ..., T\}$. For each position t from which the user has borrowed $\hat{y}_t^i$, she has to deposit in buy orders a collateral worth at least $\hat{y}_t^i/p_t$. User i's excess collateral in base asset is:

$$\text{EC}_i = \sum_{s=1}^{S} x_s - \sum_{t=1}^{T} \hat{y}_t^i/p_t$$

where:

- $\sum_{s=1}^{S} x_s$ is the sum of user's deposits
- $\sum_{t=1}^{T} \hat{y}_t^i/p_t$ is the sum of collaterals needed in case all user's borrowing positions are liquidated

Excess collateral is a key concept which positivity constraint determines the maximum amount a user can borrow (also conditional on the availability of assets, see next section) and the maximum amount a user can remove from a pool (this formalizes Rule 4)

Pool's excess liquidity

Not all quite assets deposited in a pool of buy orders can be borrowed, as sufficient incentives must be preserved for traders to take the non-borrowed part of the assets when the price hits the limit price. Pool i's excess liquidity (EL) is defined by

$$\mathrm{EL}_i = \sum_j D_j - \sum_h B_h$$

where D_j are total deposits in the pool and B_h borrowed assets.

Pool i's EL must always be positive. Borrowing or withdrawing from the pool reduce pool's EL and makes less assets available to borrow or withdraw for other users. Interest rate programatically increases with pool's utilization rate:

$$\mathrm{U}_i = \frac{\sum_h B_h}{\sum_j D_j}$$

so that either lenders will be able to exit the pool and users borrow from the pool or interest rate will be very high (around 100%).

Appendix B: Continuous compounding

Interest income accrues every second using the block timestamp. R_0 is the initial interest rate at date 0. n_1 seconds later, the interest rate changes to R_1, n_2 seconds later to R_2 and so forth.

The continuously compounded interest rate over the period $[0, T]$ is $e^{\mathrm{TWIR}_T} - 1$ where TWIR_T is the date T time-weighted sum of interest rates since origin:

$$\mathrm{TWIR}_T = n_1 \frac{\mathrm{R}_0}{N} + (n_2 - n_1)\frac{\mathrm{R}_1}{N} + ... + (n_T - n_{T-1})\frac{\mathrm{R}_{T-1}}{N}$$

with R_t/N the date t instantaneous rate and N the number of seconds in a year.

Suppose a user borrows assets between dates $t \geq 0$ and T (present). Borrower's interest rate is $e^{\mathrm{dTWIR}_{t\to T}} - 1$ with $\mathrm{dTWIR}_{t\to T} = \mathrm{TWIR}_T - \mathrm{TWIR}_t$. The third-order Taylor expansion is:

$$e^{\mathrm{dTWIR}_{t\to T}} - 1 \approx \mathrm{dTWIR}_{t\to T} + \frac{\mathrm{dTWIR}^2_{t\to T}}{2} + \frac{\mathrm{dTWIR}^3_{t\to T}}{6}$$

The interest rate earned by individual depositors can be tracked by applying current pool's interest rate to deposit size normalized by the pool's utilization rate. While TWIR_T applies to borrowed assets, TUWIR_T, the date T time- and UR-weighted sum of interest rates applies to deposits:

$$\mathrm{TUWIR}_T = n_1 \frac{\mathrm{R}_0}{N}\mathrm{U}_0 + (n_2 - n_1)\frac{\mathrm{R}_1}{N}\mathrm{U}_1 + ... + (n_T - n_{T-1})\frac{\mathrm{R}_{T-1}}{N}\mathrm{U}_{T-1}$$

as 1 dollar of borrowed assets is equal to 1 dollar of borrowable deposits multiplied by utilization rate B/D.

Suppose a lender deposits borrowable assets between dates $t \geq 0$ and T (present). Lender's interest rate is $e^{\mathrm{dTUWIR}_{t \to T}} - 1$ with $\mathrm{dTUWIR}_{t \to T} = \mathrm{TUWIR}_T - \mathrm{TUWIR}_t$. The third-order Taylor expansion is:

$$e^{\mathrm{dTUWIR}_{t \to T}} - 1 \approx \mathrm{dTUWIR}_{t \to T} + \frac{\mathrm{dTUWIR}_{t \to T}^2}{2} + \frac{\mathrm{dTUWIR}_{t \to T}^3}{6}$$