Hi all,

We would like to propose a development plan for the coming months for discussion.

This plan has been discussed with GMX contributors, and is prioritized based on our understanding of what would improve the protocol most, while balancing the development time required.

Before going into the plan, a brief recap of the development completed in the current year:

- GMX V2 improvements:
- Adaptive funding rate: improve long / short balancing
- Adaptive funding rate: improve long / short balancing
- RewardRouter updates:
- Capping of boost for Multiplier Points
- Distribution of rewards in GMX tokens
- Capping of boost for Multiplier Points
- Distribution of rewards in GMX tokens
- GMX V2.1
- Shifts: allow liquidity to be more easily moved between markets
- Auto-cancelling orders: auto-cancel orders on position close
- Atomic withdrawals: to better support lending markets
- Kink borrowing rate: allow borrowing rates to be lower during regular usage and to scale in times of demand
- Switched to Chainlink Data Stream time-based oracle for better speed and reliability
- Shifts: allow liquidity to be more easily moved between markets
- Auto-cancelling orders: auto-cancel orders on position close
- Atomic withdrawals: to better support lending markets
- Kink borrowing rate: allow borrowing rates to be lower during regular usage and to scale in times of demand
- Switched to Chainlink Data Stream time-based oracle for better speed and reliability
- GLV
- To automate moving of liquidity between markets, allowing more markets to be supported
- To automate moving of liquidity between markets, allowing more markets to be supported
- Integrated Chaos Labs Risk Oracle
- Allow risk parameters to be updated at a high frequency to safely match market conditions while providing better pricing for traders
- Allow risk parameters to be updated at a high frequency to safely match market conditions while providing better pricing for traders
- Ådded support for GMX token buybacks

With the integrated Chaos Labs Risk Oracle, work is currently being completed to reduce the price impact for all markets.

Work to reduce the borrowing and funding rates is also in progress.

For the coming months, we propose to work on:

GMX V2.2

- Gasless transactions
- In times of congestion, users may encounter issues with sending their transaction through their wallet

- Gasless transactions will allow users to trade by just signing a message in their wallet, so that even if the wallet is not able to send transactions due to congestion, the trade can still be broadcasted by a keeper network such as Gelato ([Gelato Network | Relay: Gasless transactions to supercharge your web3 UX | Enabling developers to get transactions validated fast, reliably and securely](#))

- This will also help to simplify the 1ct setup process as the 1ct account will no longer be required to be funded with any gas tokens

- Goal: improve stability / reliability, improve UX

- In times of congestion, users may encounter issues with sending their transaction through their wallet

- Network Cost Subsidies

- We have received feedback about high network costs during times of congestion

- This feature would allow a network fee pool to be accumulated from a portion of open / close fees

- This network fee pool would be used to subsidize a percentage of the user's network cost

- Tiers would be supported so that the percentage subsidized will vary depending on the trade size to prevent gaming or intentional wastage of this subsidy

- Support of this pool using a percentage of open / close / swap fees would require a Snapshot vote to enable

- Goal: lower network fees / improve UX for traders

- Multichain

- We propose to add support for virtual accounts which would allow users from any supported chain to trade on GMX

- Flow:

- The user would send a transaction on the source chain, this transaction would bridge funds to a current GMX deployment, e.g. on Arbitrum or Avalanche

- The bridge transaction would contain a message of who the sending user is

- The bridged funds would be stored in a MultichainVault and recorded to be under the sending user's balance

- To trade, the user would just need to sign a message on the source chain for each trade, this can be further simplified if the user enables 1ct

- To trade, the user would just need to sign a message on the source chain for each trade, this can be further simplified if the user enables 1ct

- With this flow, the user will not need to switch networks in their wallet, will not need to bridge any gas token for trading and can have full access to the existing liquidity on Arbitrum and Avalanche

- In future iterations, a flow can be set up for GM tokens to be mintable from any supported source chain, the GM token will then be bridged back to the source chain to be usable by integrations and other projects on the source chain

- Goal: allow users from any supported chain to trade on GMX utilizing existing liquidity

- We propose to add support for virtual accounts which would allow users from any supported chain to trade on GMX

- Flow:

- The user would send a transaction on the source chain, this transaction would bridge funds to a current GMX deployment, e.g. on Arbitrum or Avalanche

- The bridge transaction would contain a message of who the sending user is

- The bridged funds would be stored in a MultichainVault and recorded to be under the sending user's balance

- To trade, the user would just need to sign a message on the source chain for each trade, this can be further simplified if the user enables 1ct

- The user would send a transaction on the source chain, this transaction would bridge funds to a current GMX deployment, e.g. on Arbitrum or Avalanche

- The bridge transaction would contain a message of who the sending user is

- The bridged funds would be stored in a MultichainVault and recorded to be under the sending user's balance

- To trade, the user would just need to sign a message on the source chain for each trade, this can be further simplified if the user enables 1ct

- With this flow, the user will not need to switch networks in their wallet, will not need to bridge any gas token for trading and can have full access to the existing liquidity on Arbitrum and Avalanche

- In future iterations, a flow can be set up for GM tokens to be mintable from any supported source chain, the GM token will then be bridged back to the source chain to be usable by integrations and other projects on the source chain

- Goal: allow users from any supported chain to trade on GMX utilizing existing liquidity

- Cross-collateral

- This feature would enable using e.g. USDC as collateral to trade in the single token ETH [WETH-WETH] pool

- Goal: better utilization of liquidity, improve flexibility for traders and liquidity providers

- This feature would enable using e.g. USDC as collateral to trade in the single token ETH [WETH-WETH] pool

- Goal: better utilization of liquidity, improve flexibility for traders and liquidity providers

- Capped price impact

- Based on the ideas from Q

- Instead of charging price impact on position open, the open position price impact can be stored in the contract and the net price impact (open + close price impact) can be charged on position close

- This would allow price impact rebates to be applied to an entire trade

- For example, if the price impact cap for BTC and ETH is set to zero, this would lead to price impact being virtually zero for these markets

- While price impact is deducted upfront, the price impact will later be rebated if no price manipulation was detected

- Current price impact rebates require a manual review process, which leads to the time required for claiming to be between 3-10 days

- The Chaos Labs Risk Oracle could potentially support automating the flagging of price manipulation to reduce this time needed to 3 days

- While price impact is deducted upfront, the price impact will later be rebated if no price manipulation was detected

- Current price impact rebates require a manual review process, which leads to the time required for claiming to be between 3-10 days

- The Chaos Labs Risk Oracle could potentially support automating the flagging of price manipulation to reduce this time needed to 3 days

- For markets that may not be as liquid as BTC and ETH, some price impact would likely be required but can still be capped depending on the parameters recommended by Chaos Labs

- Goal: reduce price impact for traders

- Based on the ideas from Q

- Instead of charging price impact on position open, the open position price impact can be stored in the contract and the net price impact (open + close price impact) can be charged on position close

- This would allow price impact rebates to be applied to an entire trade

- For example, if the price impact cap for BTC and ETH is set to zero, this would lead to price impact being virtually zero for these markets

- While price impact is deducted upfront, the price impact will later be rebated if no price manipulation was detected

- Current price impact rebates require a manual review process, which leads to the time required for claiming to be between 3-10 days

- The Chaos Labs Risk Oracle could potentially support automating the flagging of price manipulation to reduce this time needed to 3 days

- While price impact is deducted upfront, the price impact will later be rebated if no price manipulation was detected

- Current price impact rebates require a manual review process, which leads to the time required for claiming to be between 3-10 days

- The Chaos Labs Risk Oracle could potentially support automating the flagging of price manipulation to reduce this time needed to 3 days

- For markets that may not be as liquid as BTC and ETH, some price impact would likely be required but can still be capped depending on the parameters recommended by Chaos Labs

- Goal: reduce price impact for traders

- Net OI

- Based on the same post above from Q

- We propose to add a configuration to restrict the max difference between the long and short open interest

- This restriction would apply only on position open and not on position close

- For example, if there is $100m of liquidity in a GM pool and the max difference is set to $60m, potentially reserve factors for the pool could be increased from 1 to e.g. 1.5, so that $150m of open interest would be supportable by $100m of liquidity, while minimizing the exposure of the pool through the capped net open interest

- A few cases can be considered in evaluating the practicality of this change:

- Longs are in profit, shorts are in profit

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- Longs are in profit, shorts are in loss

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the

profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- Longs are in profit, shorts are in profit

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- Longs are in profit, shorts are in loss

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- If the supportable open interest can be increased, then the borrowing fee can be reduced even in times of high open interest, while still creating a significant APR to incentivise liquidity into the GM pool

- Goal: increase liquidity efficiency

- Based on the same post above from Q

- We propose to add a configuration to restrict the max difference between the long and short open interest

- This restriction would apply only on position open and not on position close

- For example, if there is $100m of liquidity in a GM pool and the max difference is set to $60m, potentially reserve factors for the pool could be increased from 1 to e.g. 1.5, so that $150m of open interest would be supportable by $100m of liquidity, while minimizing the exposure of the pool through the capped net open interest

- A few cases can be considered in evaluating the practicality of this change:

- Longs are in profit, shorts are in profit

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- Longs are in profit, shorts are in loss

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- Longs are in profit, shorts are in profit

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- In this case regular ADL would apply if the net PnL increases too much relative to the pool size

- Longs are in profit, shorts are in loss

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- If longs close, the GM pool would reduce in size to payout the profits, and there would be pending losses from the shorts to cover this payout, in this case the pool is not significantly exposed

- If longs close and after that price changes such that shorts change from being in loss to being in profit, then ADLs would apply

- If shorts close, the GM pool would increase in size, which would help to back pending profits from longs, in case the profits of longs increase much more and insufficient shorts are opened to hedge the profits, then ADLs would apply

- If the supportable open interest can be increased, then the borrowing fee can be reduced even in times of high open interest, while still creating a significant APR to incentivise liquidity into the GM pool

- Goal: increase liquidity efficiency

It will take a few months to complete the changes in GMX V2.2 and the changes may be released in separate phases depending on the timeline and what can be completed.

GMX V2.3

After V2.2, we have a few ideas for V2.3:

- Cross-margin: a frequently requested feature, the support for cross-collateral in V2.2 should help allow this feature to be created

- Market groups: currently liquidity in similar perp markets are separated by pools, e.g. ETH [WETH-USDC], ETH [WETH-WETH]

- This can complicate the trading experience as there would be different pools with different funding rates, etc

- It may be possible to support market groups which would group all similar perp markets under a single group, for example there can be a ETH/USD group which would have the pools:

- ETH [WETH-USDC]

- ETH [WETH-WETH]

- ETH [USDC-USDC]

- ETH [WETH-USDC]

- ETH [WETH-WETH]

- ETH [USDC-USDC]

- This would combine perp liquidity in all the pools of a group, so that traders would not need to choose between pools

- Deposits / Swaps / Withdrawals would still be through individual pools as before

- With this UX simplified, there can be more options for liquidity providers to provide liquidity in the token which they prefer

- This can complicate the trading experience as there would be different pools with different funding rates, etc

- It may be possible to support market groups which would group all similar perp markets under a single group, for example there can be a ETH/USD group which would have the pools:

- ETH [WETH-USDC]

- ETH [WETH-WETH]

- ETH [USDC-USDC]

- ETH [WETH-USDC]

- ETH [WETH-WETH]

- ETH [USDC-USDC]

- This would combine perp liquidity in all the pools of a group, so that traders would not need to choose between pools

- Deposits / Swaps / Withdrawals would still be through individual pools as before

- With this UX simplified, there can be more options for liquidity providers to provide liquidity in the token which they prefer

After discussion with the community, these proposed plans can be compiled into a substack post.