MANTIS Whitepaper

Composable Foundation

May 28, 2024

Abstract

MANTIS (Multi-chain Agnostic Normalised Trust-minimised Intent Settlement) is the first protocol designed to be a vertically integrated intent pipeline, complete with expression, execution, and settlement. It recognises that cross-domain intent settlement interfaces closely with cross-domain Maximal Extractable Value (MEV), an under-explored space. The thesis is that interoperability between blockchain domains provides a widened solution space which introduces significant value of choice manifested in the form of cross-domain MEV and better outcomes for the user.

Previously, the ability to combine verifiable multi-domain execution with credible commitments was not possible. However, due to the cross-ecosystem Inter-Blockchain Communication (IBC) Protocol, we now have a transport mechanism that is robust enough to enable the creation of a protocol like MANTIS.

MANTIS is an intent settlement platform that functions as a multi-domain execution layer, with the view that choice creates value in a multi-domain world. Our objective is to understand new forms of cross-domain MEV and create a democratised market for expressing preferences across blockchain domains.

Keywords: Cross-Domain Intents & Cross-Domain MEV Execution Layer

Contents

1 Introduction

MANTIS (Multi-chain Agnostic Normalised Trust-minimised Intent Settlement) is a protocol designed to allow trust-minimised cross-chain user intent settlement. It offers the following features, addressing limitations of existing systems:

- It provides an intuitive interface of *intents* where user can specify their desired outcome for cryptocurrency transactions in simple terms. For example, an intent could be "I want to trade X for Y". MANTIS automatically determines the best procedure (e.g. settlement route) to reach the goal specified by an intent. This includes bridging and interacting with account abstraction protocols, decentralised finance (DeFi) applications, and gaming protocols.

- It abstracts away complexities of cross-domain communication. This offers intuitive execution of transfers as well as allows trades to be performed on different chains when this provides the user with a better outcome.

- It secures user funds by stakes of participants who handle those funds. This minimises trust by eliminating any centralised entity which can negatively influence a user's goals.

- Whenever possible, it offers Coincidence of Wants (CoW) matching across blockchains, further maximising users' returns by eliminating fees associated with transfers and trades otherwise needed to meet users' goals.

- It offers democratised revenue sharing for its participants such as block producers, validators, searchers and relayers. It creates competition and thus provides incentives for each participant to take a fair share of the framework's fees.

The remainder of the paper is organized as follows:

First, the assumptions and rationale for creating the MANTIS framework are discussed in 1.1 Thesis and 1.2 Motivation. The overall structure of this framework is then outlined in 1.3 MANTIS Architecture. Next, the core components of this framework are explored: 2 Intent Supply Chain, 3 the Inter-Blockchain Communication Protocol, 4 the MANTIS SDK, 5 the MANTIS Rollup and Intent Mechanism, and 6 Cross-Domain Expressiveness and Atomicity. Finally, 7 Conclusions are made about the overall impact of MANTIS as well as future areas for exploration and development.

1.1 The Thesis

Offering multi-domain execution widens the solution space and introduces significant value of choice. That choice manifests itself in the form of a cross-domain maximal extractable value (MEV) that is passed, at least in part, onto the user.

Until recently, there has been no way to combine verifiable multi-domain execution with credible commitments. Now, the Inter-Blockchain Communication (IBC) Protocol [15] provides a sufficiently robust communication mechanism for MEV and intent expression across various blockchain domains.

Using these tools, MANTIS enables cross-domain intent settlement as well as a new market for corresponding credible commitments (e.g. commitments to behave a certain way that are credible because the committer is sufficiently incentivised), searching (e.g. monitoring for MEV opportunities), and block building (e.g. constructing blocks, especially in a way that enables MEV extraction). This new mechanism is also powered by a secondary market for efficient blockspace. Users of the system specify generalised intents in an expressive manner. Solvers then compute the best possible solution based on the user's parameters, while searchers condition themselves based on the transactions that are going to different domains, and block builders/relayers form commitments with other actors in the ecosystem to ensure block inclusion. The overall mechanism abstracts away complexity for users interfacing with protocols, searchers interacting with multiple domains, and block builders obtaining a new source of order-flow.

1.2 Motivation

Pairing user experience with optimised execution has been the missing piece needed to unlock both capital efficiency and value accrual for participants in the DeFi transaction supply chain.

Alas, multi-chain interactions have continued to be rotational with users moving from one ecosystem to another or centralising [1] with off-chain trusted actors. The rise in significant centralisation within the multi-chain pipeline has continued not only in the bridging architectures themselves, but also in market makers who are filling bridging transactions.

These centralised mechanisms have been the typical methodology to address intents in the DeFi space. This is counter-intuitive to the crypto-ecosystem that typically strives for on-chain provability. However, there is a lack of decentralised solutions that rival existing centralised structures in terms of speed or cost.

This shortcoming is being addressed with the launch of trust-minimised bridging technologies, such as Picasso Network [4], that allow for generalised message passing as well as the synchronisation of multiple outlets within the DeFi landscape via the IBC Protocol.

Intents are another novel solution in the DeFi space, and have been hailed as an opportunity to revolutionise the user experience. For instance, as of April 2024, UniswapX has processed around $9 billion [8], and CoW Protocol around $2.3 billion in notional volume [7], demonstrating their significant impact and adoption in the market.

The presently discussed framework recognises that inherently there is more than one solution for any single user intent, with protocols themselves being "solutions" in addition to off-chain actors. Availability of different execution pathways and the difference between their results and the user's desired outcome is chain-dependent. This chain-dependence (i.e. executing in one domain vs. another) should be something that is offered to the user as a participant in the ecosystem.

Upcoming solutions like Anoma or Intent Essential [17, 9] aim to solve some coordination problems surrounding intents. However, they leave a significant amount of potential value that could flow to the user by not exploring the problem space by tapping into multiple different chains and protocols. Furthermore, by not being vertically integrated with execution and settlement solutions, these protocols are not able to accrue value from payment for orderflow (PFOF). We see MEV and PFOF as complementary, and eventually, these mechanisms will become the solution that allows users to transact in a heavily subsidised manner.

By leveraging the growing reach of the IBC Protocol [22], MANTIS coordinates actors operating in different domains (some on-chain, some off-chain) to realise better execution to the end user and higher MEV to participants of the system. All of this is accomplished while maintaining trust minimisation and on-chain proofs.

1.3 Architecture

The MANTIS architecture is designed with three core pillars:

- A submission layer to a Solana Virtual Machine (SVM) rollup with settlement that facilitates intent submission and counterparty discovery.

- A cross-domain auction mechanism that facilitates efficient blockspace allocation and atomicity.

- A commitments mechanism between chains that allows these conditions in the other parts of the architecture to be carried out cross-domain.

The architecture relies on two core pieces of infrastructure, provided by Picasso [4]:

- IBC, which is the mechanism coordinating work between chains. This work involves communicating execution plans as well as forming enforceable commitments between different blockchains.
- A restaking pool that coordinates the agents that have a combination of stake in various chains. Commitments formed between these actors draw upon this restaking pool.

2 Intent Supply Chain

Intents express what a user wants to achieve when they interact with a blockchain protocol. Practically, an intent is an off-chain message that encodes state transitions the user wants to achieve. Unlike transactions (which specify exact steps to take), intents are flexible. They describe a general goal but need other direct or indirect operations in order to form a final balanced transition that satisfies all of the user's constraints. Intents can be conceptualised as a means by which users express their preferences across various domains without specifying the methods to achieve desired outcomes.

In practice, an intent is a signed message that specifies a basket of assets the user is willing to spend (or, the *initial state*) and a utility function (or, the *preference map*) over a set of assets defining a user's desired goal. The assets may live on different blockchains or the user may specify a chosen final blockchain and address. For example, a transfer intent specifies some asset X on blockchain $\mathcal{B}_1$ as the initial basket of assets and a utility function assigning greater weight to the same asset on another blockchain $\mathcal{B}_2$. Or, a limit order specifies some asset X as the initial state with a utility function which assigns weight to another asset Y based on the worst-case execution price user is willing to accept. Preference maps can be even more general than that, and express other types of orders such as stop-loss orders or rebalancing orders to hedge different portfolio strategies.

2.1 Intents and MEV

The MANTIS framework facilitates a network of competing solvers that devise solutions aimed at optimising user results. Both the expression of intents and the solutions provided by solvers often create extractable value for other parties, commonly referred to as MEV [6]. For instance, the resultant solutions can change Automated Market Maker (AMM) reserves, thereby potentially creating opportunities for sandwich attacks, arbitrage, or other forms of value extraction strategies. Consequently, intents may have a relationship with MEV. In scenarios where intents are confined to a single domain, various solutions such as CoW Protocol [5] have been proposed to ensure efficient execution. Moreover, in such single-domain settings, users, solvers, and searchers can achieve atomicity, thereby reducing risks for searchers and enabling them to express their preferences more efficiently.

Cross-chain intents and transactions, which involve multiple blockchains with varying consensus mechanisms, block times, and security models, present unique challenges. The absence of standardisation and atomicity as well as the presence of delays associated with transfer bridges complicates the execution for solvers and increases their risks. Each segment of the execution process is vulnerable to front-running and significant price fluctuations. Consequently, solvers must incorporate these risks into their pricing models. Moreover, there is a notable deficiency in the ability to synchronise transactions across different chains. **Although recent advancements have been made in implementing preconfirmations at both L1s and L2s, the comprehensive capability for synchronous composability—essential for expressing cross-domain preferences atomically—remains unsolved.**

3 The Inter-Blockchain Communication Protocol

IBC is an end-to-end protocol for reliable and authenticated communication between blockchains [15]. It enables bi-directional message passing between two blockchains within a relatively short time window (an average of less than one minute per transmission between Cosmos chains, for example) [19].

Before two blockchains can communicate along this protocol, an IBC connection must be established. This is done through a handshake whose purpose is to verify the identity and status of each chain [16]. Once the connection is established, smart contracts on each chain can begin exchanging packets.

3.1 IBC Communication

As part of IBC, each blockchain runs a light client [24] of the counterparty ledger. To guarantee secure communication, the light clients are able to verify each other's blocks and state.

The method of block verification depends on the consensus algorithm used. For commonly used Proof of Stake (PoS) schema, the light client maintains a list of trusted validators and verifies new blocks by checking if enough validators have signed it[1]. Once a block is accepted, a light client's state is updated, potentially refreshing the trusted validator set.

Once the light client knows the block of the counterparty, it needs to be able to verify its state. This is typically done via Merkle proofs [23].

With those primitives, IBC communication operates as follows:

1. A smart contract on chain $\mathcal{B}_1$ sends a packet addressed to chain $\mathcal{B}_2$. The packet (technically just its commitment in the form of a hash) is stored in the state on chain $\mathcal{B}_1$.

2. An off-chain relayer sends chain $\mathcal{B}_1$'s new block to the light client running on chain $\mathcal{B}_2$. The light client verifies the block and thus gets an updated state commitment.

3. An off-chain relayer sends a membership proof of the packet stored in chain $\mathcal{B}_1$'s state to the light client on chain $\mathcal{B}_2$. The light client can verify the proof and confirm that the packet has indeed been sent from ledger $\mathcal{B}_1$. With that knowledge, the light client delivers the message to a receiver on ledger $\mathcal{B}_2$.

Crucially, because the off-chain relayer includes verifiable proofs of all data it sends to the light clients, it does not need to be trusted.

3.2 IBC Security

IBC offers trust-minimised communication between two ledgers. Any attempt at fraud is thwarted by the proof that all cross-blockchain messages include. This allows participants running on different chains to cooperate without putting trust in a third party to facilitate communication between the actors. Furthermore, any statement made by a participant can be verified by anyone, such that participants a) cannot renege on their commitments and b) can be sure that any rewards they have been promised will be delivered.

In addition, the membership and non-membership proofs serve as a fraud-proof layer. Any party (be it participating in the system or just observing from sidelines) can verify and validate actions taken by any of the actors. These actions can then be compared to requested operations to verify whether they follow the protocol or misbehave. Thanks to IBC's secured communication and state proof implementation, this verification can be performed even across chains.

[1] The initial trusted set is gathered and confirmed when a connection is established. IBC employs a procedure which allows each side of the connection to verify the identity of the chain on the other side. However, full details of the IBC Protocol are beyond the scope of this document. It is sufficient to understand that this procedure is secure and based on membership and non-membership proofs.

4 MANTIS SDK: A Chain Abstract Execution Framework

The MANTIS SDK defines a communication and packet format which allows smart contracts on one chain to execute arbitrary instructions on a different chain. It is built on top of IBC, allowing authenticated and trustless communication between the ledgers.

IBC and its extensions defined in Inter-Chain Standards (ICS) allow for a limited set of cross-chain operations. Specifically, the base protocol supports message passing between modules on chains and ICS-20 [14] introduces the additional feature of transferring fungible tokens between ledgers.

While this covers a lot of the use cases for IBC connections, if an operation does not have a corresponding ICS or a blockchain does not implement it, a smart contract on one chain is limited in what actions it can take on the other chain.

The MANTIS SDK addresses that shortcoming by defining a simplified virtual machine which can execute arbitrary instructions sent through an IBC channel.

For example, suppose that a user holds token X on a blockchain $\mathcal{B}_1$ and wants to trade it for token Y. While it may be possible to make that transaction on $\mathcal{B}_1$, it may not provide the user with the best execution. Let us say that exchange on block $\mathcal{B}_2$ has a much better exchange rate. With a pure IBC solution, the user would need to transfer the token to $\mathcal{B}_2$, sell it, and then transfer it back to $\mathcal{B}_1$.

With the MANTIS SDK, it is possible to transfer the token to $\mathcal{B}_2$ together with a MANTIS SDK program which will automatically send it to the exchange and then transfer token Y back to $\mathcal{B}_1$. Leveraging IBC's fraud-proof design, the operation is trustless and verifiable such that the moment the packet is sent from $\mathcal{B}_1$, the user can be sure that they will get their funds back.

This enables processing user actions from a single wallet and powers chain-agnostic execution.

5 Rollup and Intent Mechanism

The MANTIS rollup operates as a virtual machine-based rollup on the Solana blockchain. Specifically, it is a Solana Virtual Machine (SVM) rollup. Its primary role is to serve as a coordination and settlement layer for cross-domain, intent-based mechanisms. MANTIS, together with IBC and the MANTIS SDK, offers robust infrastructure that simplifies the design of cross-domain decentralised applications. These applications are defined by key components: a) scoring, b) solver participation, c) solution settlement, and d) cross-domain integrity proofs and proofs of misbehaviour. Additionally, the MANTIS SDK 4 is available for building these mechanisms and chain-abstract applications, enhancing the development and integration process for developers.

The MANTIS rollup provides a framework for block proposals and credible commitments across various domains. These commitments are held accountable through the IBC security model, ensuring transparency and security in multi-chain interactions.

The rollup is decentralised, featuring a rotating set of independent sequencers in a round-robin fashion to ensure a censorship-resistant environment. This decentralisation is crucial for guaranteeing that all intents and solutions are accepted before the end of the scoring period, thereby maintaining credibility in scores and facilitating the blockspace auctions discussed in section 6.

The architecture of the MANTIS rollup is depicted in Figure 1, and further described throughout this section of this paper.

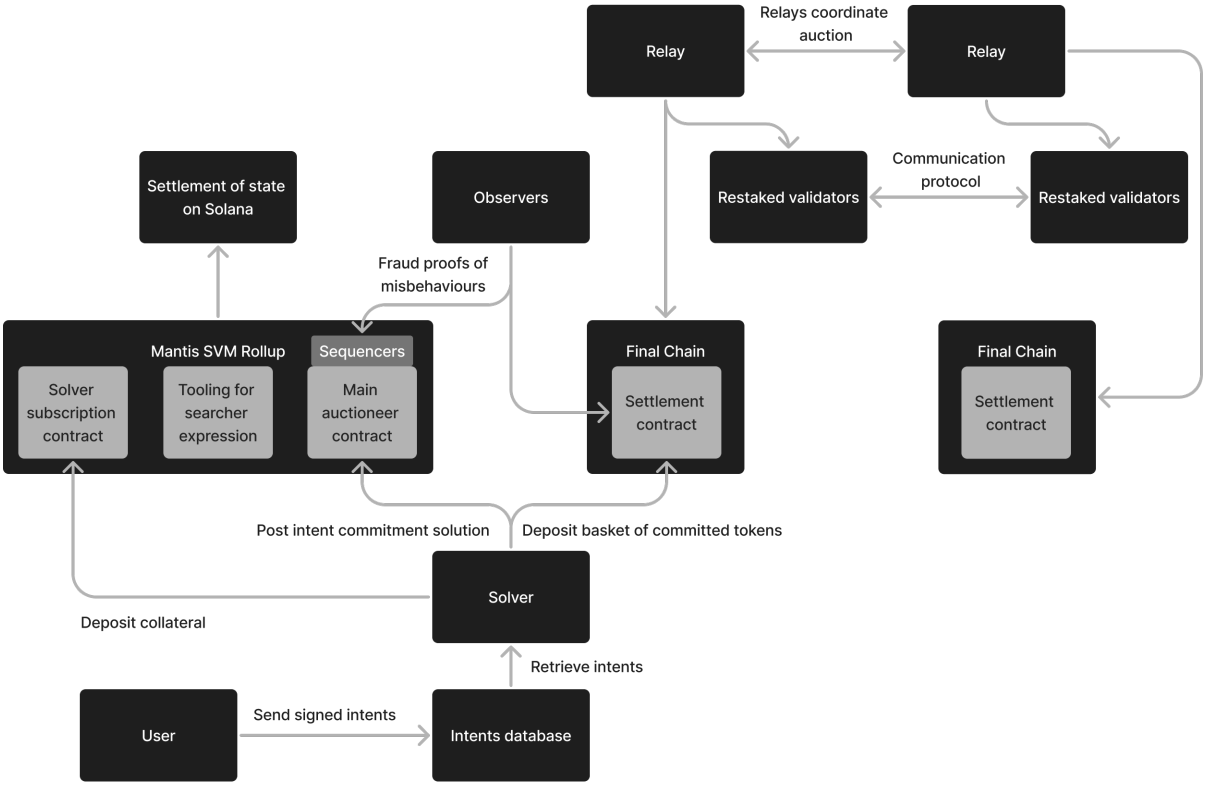


Figure 1: MANTIS rollup architecture.

5.1 Bridge Contract

The flow of tokens through the MANTIS rollup is depicted in Figure 2. This flow is facilitated by a bridge contract that interacts with the MANTIS rollup. Users are able to deposit Solana (SOL) tokens as well as a number of liquid staked tokens (LSTs) from SOL into this bridge contract.

Then, any deposited SOL tokens are staked with Solana validators for Proof of Stake (PoS) validation. This staked SOL will be represented by an LST. Users can opt to have this LST restaked on the backend into the Restaking Layer of Picasso [4]. This provides users with increased staking yield while abstracting away the additional steps for users to perform restaking on their own.

Any LSTs that have been directly deposited into the bridge contract flow to the restaking vault. The resulting crypto-economic security can be leveraged by the MANTIS rollup as well as Actively Validated Services (AVSes) paying for this security.

Additionally, when deposited into the bridge contract, stablecoins that are Solana native will be deposited into lending protocols on Solana such as Kamino [18] and marginfi [21]. The user can then choose to restake their stablecoins into the Solana restaking layer. This also results in increased yield for the user.

If a user wants to withdraw their (re)staked tokens, a fraud proof is generated to ensure that the sequencer has not misbehaved. This is a specific zero knowledge (ZK) proof that unlocks funds on the bridge contract and contains a MANTIS SDK instruction. Once this proof goes through, funds are unlocked and sent back to the user in the form in which they were deposited (i.e. if a user deposited SOL, SOL will be returned to them). The user can then bridge these tokens to other locations or perform other functionalities with them again over IBC.

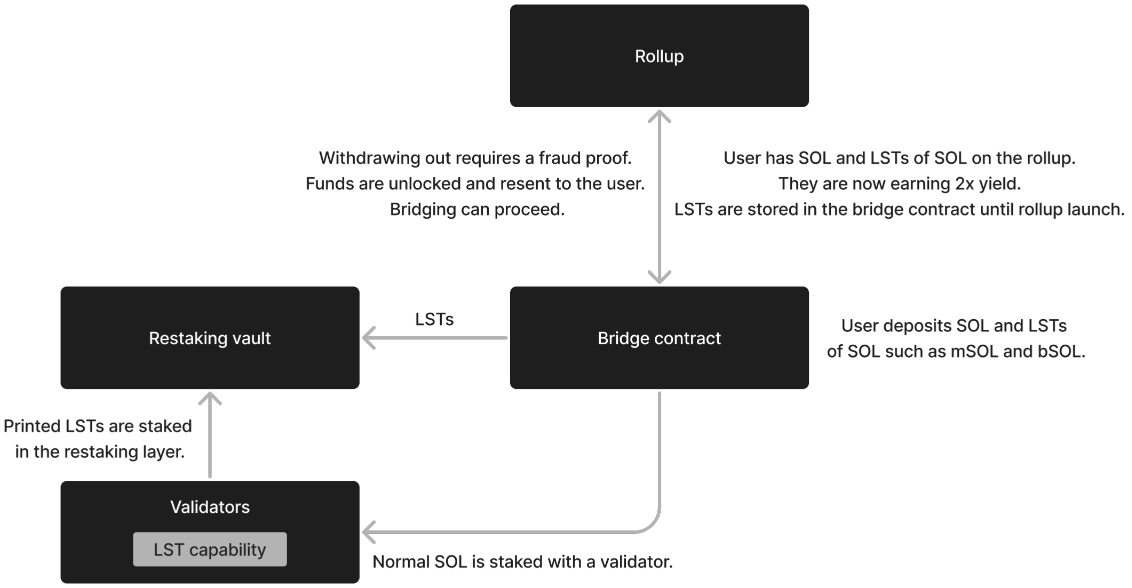

Rollup
Withdrawing out requires a fraud proof.
Funds are unlocked and resent to the user.
Bridging can proceed.
User has SOL and LSTs of SOL on the rollup.
They are now earning 2x yield.
LSTs are stored in the bridge contract until rollup launch.
Restaking vault
LSTs
Bridge contract
User deposits SOL and LSTs
of SOL such as mSOL and bSOL.
Printed LSTs are staked
in the restaking layer.
Validators
LST capability
Normal SOL is staked with a validator.

Figure 2: MANTIS rollup bridge contract flow.

5.2 Intent Mechanism

MANTIS consists of a mechanism for optimal execution of cross-domain intents via a competition of solvers. A summary of the protocol goes as follows:

Users sign intents which are contained on a private rollup mempool. Solvers are staked agents that can a) observe the transactions on the mempool and b) post solutions in the auctioneer contract of the rollup. The auctioneer contract scores the solvers' solutions in terms of users utility maximisation. The winner of the auction is responsible for settling the outcome of the intent to the solution settlement contracts in the final chain expressed by the intent. This settlement is completed by sending the transaction out over IBC to the involved protocols and blockchains.

The intent mechanism key components thus consist of the auctioneer, solvers, the main auctioneer contract, the solution settlement contracts, and the staking contract.

Solvers are the parties responsible for computing and submitting *intents solutions* (or simply *solutions*). A solution constitutes of a specific feasible path for an intent to be settled. For example, in the case of a swap between two chains, the solution could be a route that consists of trading in different AMMs as well as the necessary IBC transfers between domains of these AMMs. Another solution could consist of solvers providing private liquidity to the destination chain.

The **auctioneer** is the party (or set of parties) responsible for a) storing the mempool (intent database) and b) allowing solvers to observe non-executed intents.

The **main auctioneer contract** is a rollup component where solutions that satisfy the feasibility constraints are settled and ordered according to the intent preference map. These constraints include a) the conditions specified by the intent preference map and b) the requirement that the value staked by the solvers exceeds the intent's worst execution costs. Once the time for posting solutions concludes, the winner of the auction can unlock the tokens of the intent being solved. This main auctioneer contract is also responsible for slashing possible misbehaviour of solvers via proof of misbehaviour and repaying users when execution of their intent fails.

The **solution settlement contracts** on the destination chains consist of all contracts where solutions are executed and a state proof of the execution of the intent is stored. In case of a dispute due to a solver misbehaving, any party can submit a non-membership proof of the intent solution on MANTIS roll-up via IBC.

To participate in this mechanism, solvers stake tokens to the **staking contract** and are required to make periodic payments to access the intent database. This payment secures the exclusive right to retrieve intents, ensuring that solvers are financially committed to participating in the auction process.

Slashing conditions depend on the specifics of the intents. In the case that the intents are swaps, the slashing conditions are applied under two primary scenarios:

1. If the solution is not posted on the final chain, the slashing penalty depends on the price movement of the underlying assets associated with the intent. In the absence of significant price movement, a fixed slashing amount is proposed. However, if substantial price movement is observed, the slashing penalty mirrors the condition described in the subsequent scenario.

2. If the solution executed does not match the solution committed in the main auctioneer contract, the solver is slashed proportionally to the difference between the committed amount and the actual amount provided to the user. This condition ensures that solvers are held accountable for any discrepancies in their commitments versus deliveries.

A dedicated network of observers is responsible for monitoring potential misbehaviour among participants. This network actively sends an IBC proof of misbehaviour to the main auctioneer contract when misbehaviour is detected. Observers are incentivised with rewards for their vigilance and reporting, which promotes a robust monitoring system and enhances the integrity of the entire auction process.

As a summary, the intent lifecycle proceeds as follows. First, the user expresses and signs an intent, sending it to the intent database in the MANTIS rollup. For a period of time, solvers propose different solution intent commitments to the auction smart contracts. The solutions are then scored, taking into account the intent preference map. Once the winning solution is picked, the corresponding solver must execute the solution on the destination chain. If the solver misbehaves, an IBC proof of misbehaviour is generated and sent to the auctioneer contract, resulting in the slashing of the misbehaving solver. This overall process is depicted in Figure 3.

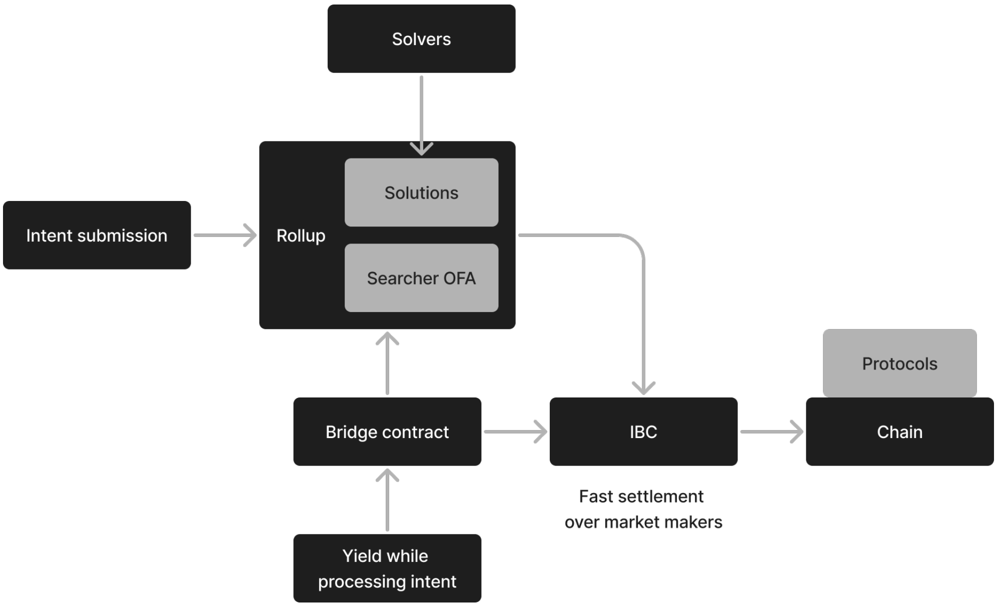


Figure 3: Intents submission and settlement.

Figure 4 further depicts how intent settlement works cross-chain. Intents can be submitted from IBC-connected chains (such as Solana or Ethereum) or protocols (such as Osmosis, a

Cosmos-based decentralised exchange). Regardless of origination location, all intents are sent through the MANTIS rollup, where solvers are able to create solutions using CoWs or requests for quotes (RFQs) to market makers to provide liquidity. The resulting transaction routes are processed over IBC, where they are able to be settled on any IBC-enabled destination chain or protocol.

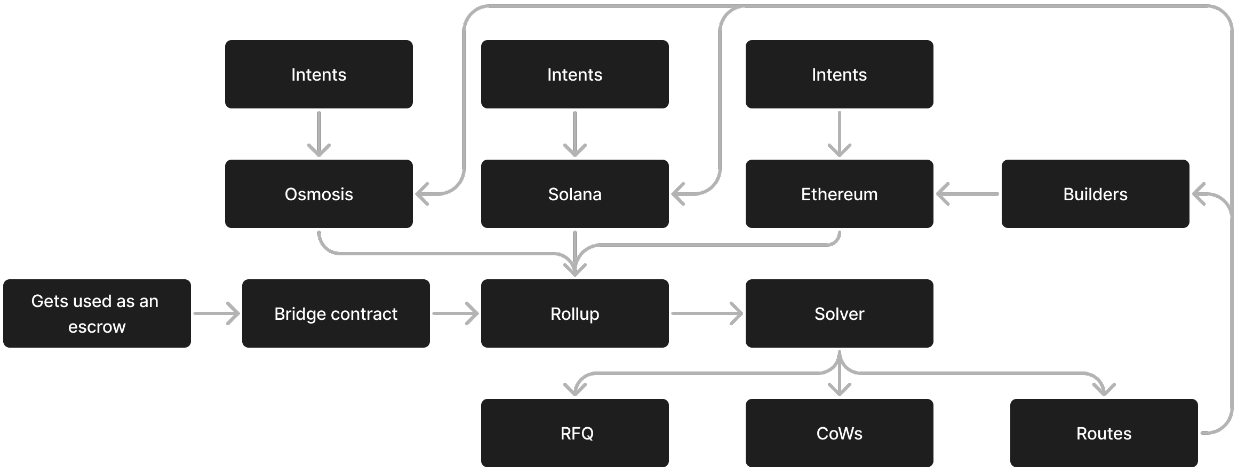


Figure 4: Cross-chain intents submission and settlement.

6 Cross-Domain Expressiveness and Atomicity

Pricing cross-domain intent solutions presents multiple challenges for solvers. As [3] shows, these challenges are primarily financial. Solvers must first manage entry costs, which encompass the initial expenses related to setting up infrastructure necessary for participating in intent markets. Additionally, congestion costs arise when entering the market; these costs vary depending on the number of competitors and affect solvers while they vie for opportunities. Finally, solvers contend with the risk of non-atomic execution due to the lack of expressiveness of blockspace allocation for multiple chains, further complicating their operational landscape. This underscores the need for mechanisms that mitigate these latter costs for solvers, potentially enhancing overall user welfare. Moreover, the existence of such mechanisms will reduce the cost of MEV extraction between chains, increasing the revenue of validators, as has been seen in single-domain MEV auctions.

Already, single-domain MEV auctions have commoditised the single-domain MEV opportunities such as sandwiching and arbitrage between decentralised exchanges (DEXes) by minimising the risks that searchers are exposed by extracting MEV opportunities and reducing entry costs. As a consequence, validators' revenue increased by efficiently allocating blockspace to the searchers that value it the most through a sealed bid first-price auction. Moreover, MEV auctions enable users to send their transactions in a manner that a) has pre-execution privacy via a private communication channel between user and builder and b) removes the possibility of reverted transactions. This increases users' welfare by reducing adverse selection (e.g. front-running) and the gas costs induced by reverted transactions.

Our thesis is that users, searchers, and solvers do not just have preferences over state transitions for one domain, but in general, accrue value from combinatorial state transitions over different domains. A natural example is a searcher that exploits price discrepancies of DEXes on two different domains. However, these preferences over state transition currently lack the necessary infrastructure to be expressed atomically. At most, currently users bid simultaneously on an independent auction, hoping to get all transactions executed, though there is a high risk of execution failure. [11] shows that in economic equilibrium, expressing complex valuations over a set of goods that have complementaries (i.e. other goods or services that increase their value when used or consumed together) can have very bad outcomes when these items are sold through separated simultaneous auctions. To mitigate this, MANTIS will coordinate various actors to sell blocks from different domains through a combinatorial auction allowing validators of different chains to efficiently and fairly capture the value from selling their blocks to third parties such as builders and searchers. This mechanism must be individually rational for all block proposals to participate. That is, each block proposal should at least obtain the revenue that would be obtained by not participating in the mechanism.

While these mechanisms have a similar purpose to shared sequencing, shared sequencing is designed for servicing ledgers within the same ecosystem (such as Ethereum and its Layer 2s). Thus, shared sequencing usually assumes some alignment between the agents that do not necessarily exist between validators of different Layer 1s, and so this does not address the need of an incentive-compatible mechanism on the validator set.

MANTIS combinatorial auctions will enable builders and searchers to express bundles through different domains, ensuring atomicity as well as more revenue to validators that sell their items separately.

6.1 Just-in-Time Auction

Currently, builders and searchers have no ability to express their preferences over blocks of different domains. The objective of the just-in-time auction on MANTIS is to allow builders to express atomically. The first design of this combinatorial auction for two domains will consist of two simultaneous English auctions with a unique combinatorial block take-it-or-leave it offer. Buyers will be able to place send blocks with bids for the independent blocks and combinatorial blocks. The combinatorial blocks have the following constraint:

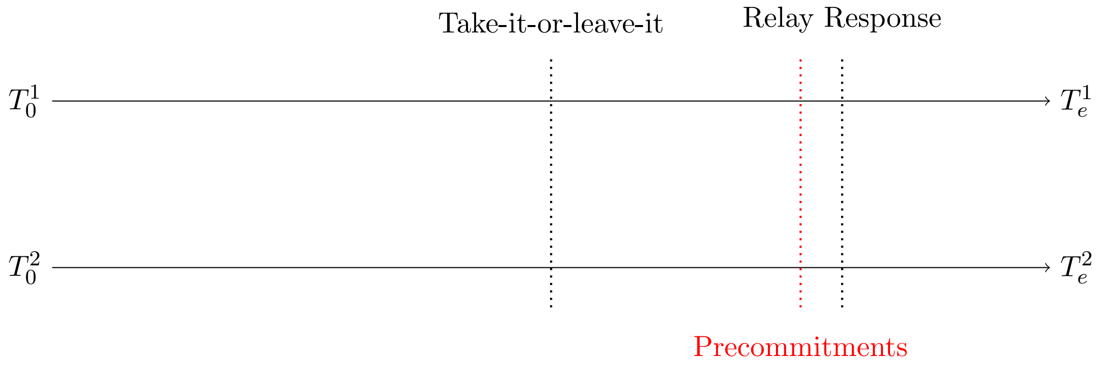
The block A with bid b_A in the first domain is accepted if and only if the block B of the second domain is accepted with bid b_B.

As a minimum viable product, first, both block proposals and the relay will have to set up a $(3, 3)$ threshold decryption scheme. The relay will allow both validators to observe the block headers, all bids submitted by independent blocks, and the counter-party block proposals pre-commitment if it is delivered. The pre-commitments contain a signature of a pre-commitment of a block-header, a share of the secret of the threshold decryption scheme, and an encrypted signature of the block-header of one of the blocks that constitutes the bundle block. When the pre-commitment phase period concludes, each validator has the option to send a pre-commitment of a bundled block. If the relay receives pre-commitments of both block proposals for the same combinatorial block before the closure of the block proposal pre-commitment period, it decrypts the signature and releases the blocks. If not, the relay sends a signed message to both parties, stating that both parties can construct their own separate blocks. More formally, given T_0 being the minimum of slots times of both domains and T_e analogously for the end time, the auction works as follows:

MANTIS auction

- Input: Bids of independent blocks and the combinatorial one.
- Output: Block allocation and payments.

1. Before the beginning of both slots T_e, both block proposals and the relay set up a $(3, 3)$ threshold decryption scheme.
2. At time $T_0 + [T_e - T_0]/2$ validators receive a take-it-or-leave-it offer taking into account the higher combinatorial bid and the bids of the independent blocks.
3. Before time $T_p = T_0 + 3[T_e - T_0]/4$, validators can send a pre-commitment message of the combinatorial block.
4. If both validators send the pre-commitment, then they immediately receive a signed message from the relay stating that the combinatorial block is committed. Then the relay decrypts the signatures and broadcasts the blocks with the take-it-or-leave-it payments to both block proposals.
5. Otherwise, the relay sends a signed message to both validators rejecting the combinatorial block and runs the standard MEV-Boost protocol [12] in this case.



This approach, however, poses two main challenges: the risk of double-signing and the high level of trust placed in the relay. The risk of double-signing is generally mitigated by the consensus protocols of the domains themselves. Nonetheless, the lucrative MEV opportunities enabled by this market may increase the likelihood of misbehaviour. Consequently, there is a need to increase the collateral staked by validators to enhance security, which we plan to address through restaking. In instances where an agent double-signs, a challenging period will be initiated in the MANTIS rollup. During this period, various observers can demonstrate that an agent double-signed. If the block proposer fails to send the relay's signed message, which authorised him to sign another block, the agent will be penalised. Conversely, if the message is sent, the relay will face penalties. The amounts for slashing should consider the costs associated with atomicity failure, missed slots, and other related expenses.

While a just-in-time auction can be close to efficient when the number of domains is small ($m = 2, 3$), all mechanisms with some desiderata constraints have very inefficient outcomes (where the inefficiency growth is at least polynomial growth with a fractional exponent in the number of domains) on equilibrium due to selfish and rational behaviour of block proposal, since agents have monopolistic veto power for not selling combinatorial blocks. Also, the computational complexity of allocating combinatorial resources [13] makes it more complex to run optimal auctions just-in-time. For this reason, we propose a future blockspace market. Future blockspace markets make future blocks fungible, decreasing the monopolistic power of sellers on selling combinations of blocks and increasing the efficiency (as an example, a double auction with full complementary goods [2]).

6.2 Combinatorial Blockspace Future Markets

The new crypto-economic primitive of restaking enables future block proposal mechanisms such as execution tickets outside of a domain's consensus protocol. Thus, restaking (such as that being facilitated by the Picasso Network [4]) provides MANTIS with a mechanism for powering more efficient blockspace allocation.

In this revised model, block proposers can issue credible commitments about future block construction, which essentially are promises to build blocks in accordance with specific conditions laid out by ticket holders if certain payment thresholds (e.g. reserve price) are met. The tickets will be exchanged via a combinatorial batch auction where buyers will be able to express combinatorial valuations over the tickets and sellers will express reserve prices. Then, these tickets can be traded or sold in a secondary market, similar to the current design of MEV-Boost and Proposer-Builder Separation [12, 10] through an approximately efficient allocation mechanism [20]. Instead of a simple lottery system for determining block proposers, the model allows for a dynamic marketplace where block space rights can be bought and sold. This outside-protocol approach permits builders to pre-confirm users' transactions in advance and also reduce the variance of validators' revenue.

Restaking also enables combinatorial tickets, and more generally, commitments' composition. These commitments, for example, allow proposers to issue more complex conditions that can depend on the actions of block proposers of other chains, enabling a market of future blockspace of different domains. The existence of such a mechanism will allow a new paradigm of cross-domain processes such as flashloans, frequent batch auctions, and simplified MEV strategies such as atomic arbitrage, potentially increasing overall user welfare and validator revenue.

The fulfilment of such commitments can be guaranteed by slashing the staked collateral of block proposals and proofs of misbehaviour enabled via the IBC Protocol.

7 Conclusions

The MANTIS framework aims to enhance cross-chain transactions and improve the efficiency of decentralised finance. The key points and future directions are outlined below:

1. **Interoperability:** MANTIS intends to facilitate smoother transactions across various domains through implementation of the IBC protocol.

2. **Economic Incentives:** The framework introduces innovative mechanisms such as combinatorial auctions and future blockspace markets, designed to provide competitive and fair economic access to combinatorial blockspace for all participants.

3. **Future Developments:** The implementation of this framework could reduce operational costs and lower barriers for users and validators, which is expected to enhance participation and liquidity within the blockchain ecosystem.

4. **Challenges and Outlook:** As MANTIS is still in the conceptual stage, addressing challenges related to managing complex cross-chain interactions and maintaining high security will be crucial as it progresses toward implementation and private intent execution.

References

[1] Across Protocol. Across protocol documentation. URL https://docs.across.to/. Accessed on 2024-05-01.

[2] Rakesh Chaturvedi and Ashish Kumar Pandey. Double auction for trading perfect complements. *Journal of Public Economic Theory*, 26(1):e12672, 2024.

[3] Tarun Chitra, Kshitij Kulkarni, Mallesh Pai, and Theo Diamandis. An analysis of intent-based markets. *arXiv preprint arXiv:2403.02525*, 2024.

[4] Composable Foundation. Picasso—the cross-ecosystem IBC & restaking hub, 2024. URL https://docs.picasso.network/. Accessed: 2024-05-01.

[5] CoW DAO. CoW Protocol documentation, 2024. URL https://docs.cow.fi/. Accessed on 23 April 2024.

[6] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In *2020 IEEE symposium on security and privacy (SP)*, pages 910–927. IEEE, 2020.

[7] Dune. Cow protocol, 2024. URL https://dune.com/cowprotocol/cowswap-high-level-metrics-dashboard?Aggregate+by_e759c2=Week. Accessed: May. 28, 2024.

[8] Dune. Uniswapx, 2024. URL https://dune.com/phu/uniswapx. Accessed: Apr. 3, 2024.

[9] Liesl Eichholz. Introducing Essential: We are intents, Jul. 2023. URL https://blog.essential.builders/introducing-essential/.

[10] Ethereum Foundation. Proposer-builder separation, Accessed: 2024. URL https://ethereum.org/en/roadmap/pbs/.

[11] Uriel Feige, Michal Feldman, Nicole Immorlica, Rani Izsak, Brendan Lucier, and Vasilis Syrgkanis. A unifying hierarchy of valuations with complements and substitutes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1), Feb. 2015. DOI: 10.1609/aaai.v29i1.9314. URL https://ojs.aaai.org/index.php/AAAI/article/view/9314.

[12] Flashbots. mev-boost, Apr. 2024. URL https://docs.flashbots.net/flashbots-mev-boost/introduction.

[13] Yuzo Fujishima, Kevin Leyton-Brown, and Yoav Shoham. Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In *IJCAI*, volume 99, pages 548–553, Jul. 1999.

[14] Christopher Goes. ICS-20: Fungible token transfer, 2019. URL https://github.com/cosmos/ibc/blob/main/spec/app/ics-020-fungible-token-transfer/.

[15] Christopher Goes. The Interblockchain Communication Protocol: An overview, Jun. 2020.

[16] Christopher Goes and Juwoon Yun. ICS-3: Connection semantics, 2019. URL https://github.com/cosmos/ibc/tree/main/spec/core/ics-003-connection-semantics.

[17] Christopher Goes, Awa Sun Yin, and Adrian Brink. Anoma: a unified architecture for full-stack decentralised applications, Aug. 2022. URL https://media.githubusercontent.com/media/anoma/whitepaper/main/whitepaper.pdf.

[18] Kamino Finance. Kamino lend litepaper, 2023. URL https://docs.kamino.finance/kamino-lend-litepaper. Accessed: 2024-05-13.

[19] Jungyeon Kim, Meryam Essaid, and Hongtaek Ju. Inter-blockchain communication message relay time measurement and analysis in cosmos. In *2022 23rd Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pages 1–6, Takamatsu, Japan, 2022. IEEE. DOI: 10.23919/APNOMS56106.2022.9919970.

[20] Brendan Lucier and Allan Borodin. Price of anarchy for greedy auctions. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 537–553. SIAM, 2010.

[21] marginfi. The marginfi protocol, 2024. URL https://docs.marginfi.com/marginfi-protocol. Accessed: 2024-05-13.

[22] Mary McGilvray. The IBC Protocol 2023 year in review, Dec. 2023. URL https://www.ibcprotocol.dev/blog/2023-year-in-review.

[23] Ralph C. Merkle. Method of providing digital signatures, 1979. URL https://image-ppubs.uspto.gov/dirsearch-public/print/downloadPdf/4309569. US Patent 4,309,569.

[24] Corwin Smith and Joseph Cook. Light clients, 2023. URL https://ethereum.org/en/developers/docs/nodes-and-clients/light-clients/.