Economics of Disputes in Arbitrum BoLD

The following document explains the economics and denial-of-service protection mechanisms built into Arbitrum BoLD. It covers trade-offs Arbitrum has to make to enable permissionless validation, explaining the key problems in an accessible way.

Background

[Arbitrum One](https://docs.arbitrum.io/welcome/get-started) is currently one of the most widely used Ethereum scaling solutions, with~14bn USD in total-value-locked at the time of writing. Not only do its scaling properties, such as its 250ms block times, make it popular, but so do its security properties and approach to decentralization. Currently, Arbitrum One is governed by the Arbitrum DAO, one of the most active and robust on-chain organizations.

However, Arbitrum technology has not yet achieved itsfull promise of being fully decentralized. Currently, withdrawals from Arbitrum One back to Ethereum are verified by a permissioned list of validators. These validators can still challenge invalid withdrawals, but the system prevents anyone outside this list from holding them accountable. This permissioned list of validators limits Arbitrum One and Arbitrum Nova to being categorized as aStage 1 Rollup on theL2Beat website , meaning it still has training wheels preventing it from reaching its full potential.

The rollup technology powering Arbitrum One is called "optimistic" because claims about its state settled to and confirmed on Ethereum after a period of approximately seven days. During those 7 days, the claimed states can be disputed. To make an analogy, a check can be cashed immediately but can be taken to court to dispute if there is a problem within a specific time frame. Because Arbitrum's state is deterministic, a validator that is running a node and following the chain will always know if a posted claim is invalid. A key decentralization property allowsanyone who knows the correct claim to challenge invalid claims andwin the challenge. This preserves the accurate history of Arbitrum settling to Ethereum and protects the integrity of users' funds and withdrawals using a "single honest party" property. As long as there is a single entity following the chain and willing to dispute a claim, Arbitrum's security guarantees are maintained.

Today, Arbitrum One's security properties are defined by the size of its permissioned set of validators. Validators could collude or finalize/confirm an incorrect state, and users have no recourse aside from the Arbitrum One security council stepping in. Elevating Arbitrum One's decentralization requires a different approach.

In the Fall of 2023, Offchain Labs announcedArbitrum BoLD , a new dispute resolution protocol built from the ground up that will bring Arbitrum chains to the next level of decentralization. BoLD, which is an acryonym forBo undedL iquidityD elay, allows permissionless validation of Arbitrum chains. This means that chain owners can remove the list of permissioned validators for their chains to allow anyone to challenge invalid claims made about Arbitrum states on their parent chain and win.

In this document, we'll explore the economics and trade-offs enabling permissionless validation.

Settling Arbitrum states to Ethereum

We frequently state that "Arbitrum chains settles their states to a parent chain", and we'll elaborate on what that means. All Arbitrum One transactions can be recreated by reading data from Ethereum L1, as compressed batches of all L2 transactions are frequently posted to Ethereum. Once a batched transaction is included in a finalized block on Ethereum, its history will never be reverted on Arbitrum One. Ethereum, however, does not know if a batch posted to it refers to a correct version of Arbitrum One's history. To verify batch integrity, there is a separate process that confirms batch correctness on Ethereum: it is called the "assertion."

For Arbitrum One specifically, approximately every hour, entities known as validators check the correctness of batches by following the[Arbitrum chain](https://docs.arbitrum.io/welcome/get-started) . Validators then post something called an "assertion", which attests to the validity of a batch, saying, "I have verified this batch". As Ethereum does not know what is correct on Arbitrum One, it allows about seven days for anyone to dispute one of these assertions. Currently, there is a permissioned list of validators who can post assertions and challenge assertion for all Arbitrum chains. Arbitrum BoLD will enable any chain owner, such as the ArbitrumDAO, to remove this permissioned list. Note that validators who opt to posting assertions are otherwise known as a "assertion proposers".

Withdrawing assets back to Ethereum from Arbitrum

Users of Arbitrum One that have bridged assets from Ethereum can withdraw said assets at any time. However, for this withdrawal to be fully executed, its corresponding claim must match a confirmed assertion on Ethereum. For instance, if Alice starts a withdrawal transaction on Arbitrum One, it gets posted in a batch on Ethereum. Then, a validator will post an assertion about that batch on Ethereum an hour later. The assertion has a seven-day window in which anyone can dispute it. If this assertion isn't disputed within that time frame, it gets confirmed. After that window passes, Alice will receive her withdrawn assets on Ethereum and is free to use them as she pleases.

"Settling states" and having a seven-day dispute window is crucial to ensuring assets can be withdrawn safely. Allowing anyone to dispute invalid claims and win keeps withdrawals protected by strong security guarantees without needing to trust a group of validators. This "permissionless validation" separates Optimistic Rollups from side chains.

The dispute period

The reason there is a dispute window for assertions about Arbitrum One on Ethereum is because Ethereum itselfhas no knowledge about what is correct on Arbitrum One. The two blockchains are different domains with different states. Ethereum, however, can be used as a neutral referee for parties to dispute claims about Arbitrum One. The dispute period is seven days because it is seen as the maximum period of time an adversary could delay Ethereum before social intervention, originally proposed by Vitalik Buterin. This window gives enough time for parties to catch invalid claims and challenge them accordingly.

Dispute resolution times

An actual dispute occurs if a party disagrees with an assertion on Ethereum and posts an assertion they know to be correct as a counter-claim. This creates a "fork" in the chain of assertions, requiring a resolution process. We'll get into the high-level details of how disputes are resolved later in this document.

Once an actual dispute is ongoing, it will also take time to resolve, as, once again, Ethereum has no knowledge of the correctness of Arbitrum One states. Ethereum must then give sufficient time for parties to submit their proofs and declare a winner. The new Arbitrum BoLD protocolguarantees that a dispute will be resolved within seven days so long as an honest party is present to defend against invalid claims.

As assertions have a dispute window of seven days, and disputes require an additional seven days to resolve, a dispute made at the last second woulddelay assertion confirmation to a maximum of 14 days , or two weeks. BoLD is the only dispute protocol we are aware of that guarantees this bound.

The cost of delaying withdrawals

Delaying withdrawals incurs opportunity costs and impacts user experience for users who want to withdraw their assets. In the happy case of no disputes, withdrawals already have a baked-in, seven-day delay. A dispute adds seven days to that delay. The problem is that disputes delayall pending withdrawals from Arbitrum One back to Ethereum, not just a single claim. As such,disputing a claim must have a cost for the initiator proportional to the opportunity cost they impose on Arbitrum users.

Requiring a bond to become a validator

By default, all Arbitrum nodes act as validators, monitoring the chain to verify assertions posted to the parent chain and flagging any invalid assertions. On Arbitrum One, running a validator, known as a “watchtower” node, is permissionless and requires no additional cost other than the infrastructure for the node.

Another type of validator, called a "proposer," performs additional tasks on top of their regular duties as a regular validator. Proposers compute Arbitrum states and propose assertions to the parent chain. To prevent abuse and delays in withdrawals, proposers must make a security deposit or "bond" to gain the privilege of proposing assertions. This bond can be withdrawn once their latest assertion is confirmed, ending their responsibilities as a proposer.

Arbitrum BoLD allows validators to become proposers and challengers without permission. Proposers must bondETH to propose state assertions to the parent chain. Only one proposer is needed for chain progress, allowing most validators to simply verify assertions. In case of disputes over state assertions, BoLD enables anyone to put up a "challenge bond" ofETH to dispute invalid assertions, acting as a challenger in defense of an Arbitrum chain.

Pricing bonds

Ensuring assertions are frequently posted is a requirement for Arbitrum, but at the same time, it should not be a privilege that is easily obtained, which is why the pricing of this "security deposit" is based on opportunity cost.

To be highly conservative, in a "bank run"-like scenario, the Arbitrum One bridge contains approximately 3.4B USD worth of assets at the time of writing on Oct 23rd, 2024. Assuming funds could earn a 5% APY if invested elsewhere, the opportunity cost of 1 extra week of delay is approximately 3.27M USD. Given this scenario, we recommend a bond for assertion posters to be greater than 3.7M USD.

Honest proposers can always withdraw their bond once their assertions are confirmed. However, adversaries stand to lose the entirety of their bond if they propose invalid assertions. A large bond size drastically improves the economic security of the system based on these two axes by making the cost to propose high and by guaranteeing that malicious actors will lose their entire bond when they are proven wrong by the protocol.

Given that participation in BoLD is permissionless, we recommend that the size of bonds required to participate be high enough to disincentivize malicious actors from attacking Arbitrum One and to mitigate against spam (that would otherwise delay confirmations up to 1 challenge period). High bonding values do not harm decentralization because (1) trustless bonding (or staking) pools can be deployed permissionlessly to open challenges and post assertions, and (2) any number of honest parties of unknown identities can emerge to bond their funds to the correct assertion and participate in the defense of Arbitrum at any time within a challenge. As with the current dispute resolution protocol, there are no protocol level incentives for parties who opt in to participate in validating Arbitrum One with BoLD.

While both of these bonds can be any ERC20 token and be set to any size, we recommend the use of theWETH ERC20 token & the following bond sizes for Arbitrum One:

- Assertion bonds: 3600ETH
-
    - required from validators to bond their funds to an assertion in the eventual hopes of having that assertion be confirmed by the rollup protocol. This is a one-time bond required to be able to start posting assertions. This bond can be withdrawn once a validator's assertion is confirmed and can alternatively be put together via a trustless bonding pool.
- Challenge-bonds, per level: 555/79ETH
-
    - required from validators to open challenges against an assertion observed on the parent chain (Ethereum, in the case for Arbitrum One), for each level. Note that "level" corresponds to the level of granularity over which the interactive dissection game gets played, starting at the block level, moving on to a range of WASM execution steps, and then finally to the level of a single step of execution.

These values were carefully calculated to optimize for the resource ratio (explained later) and gas costs in the event of an attack, as explained inBoLD whitepaper . This effectively means that an entity that has already put up a bond to propose an assertion does not need to put up a separate assertion bond to challenge an invalid state assertion that they observe. To be explicitly clear, the validator would still require 555ETH and 79ETH for ongoing challenges. These additional challenge bond amounts are needed to participate in the interactive dispute game (back and forth) and narrows down the disagreement to a single step of execution that is then proven on Ethereum. The 555ETH and 79ETH challenge bonds can be put together via a trustless bonding pool, and do not all have to be put up by the validator that opened the challenge. These bonds can be refunded at the end of a challenge and can also alternatively be put together by the community using a trustless bonding pool.

Centralization concerns

Requiring a high bond to post assertions about Arbitrum seems centralizing, as we are replacing an allowlist of validators with a system that requires substantial funds to participate. However,BoLD ships with a trustless bonding pool for assertion posting. That is, any group of honest parties can pool funds into a simple contract that will post an assertion to Ethereum without needing to trust each other. We believe that making it easy to pool the funds to become an assertion poster without needing trust to dispute invalid claims does not affect the safety or decentralization of BoLD.

We claim optimizing for the unhappy case is more important than the happy case. As there only needs to be one honest assertion poster, we believe it falls into the security budget of the chain to set a high bond fee in order to become a proposer. Itshould be expensive to delay Arbitrum One withdrawals, and it should also have a high barrier to entry to perform a key responsibility. As long as disputes can be made in a trustless manner, and trustless pools are available in production, we claim the security properties of assertion posting hold equally.

Resolving disputes

One of the core properties BoLD achieves is providing a fixed upper bound for dispute resolution times. This section will discuss the constraints required to achieve this from first principles.

Dispute game overview

Every game between adversarial parties needs a referee: a neutral party that can enforce the rules to declare a fair winner. Arbitrum BoLD relies on Ethereum as its referee because of its properties as the most decentralized, censorship-resistant smart contract chain in the world.

When a dispute happens about Arbitrum One assertions on Ethereum, there is a protocol for resolving them. At its core, a dispute is about the blockhash of an Arbitrum One block at a given height. Ethereum does not know which claim is correct and, instead, relies on a dispute game to be played out. The game involves different parties making claims with proof to eventually narrow down their disagreement to a single step of execution within the execution of a block, called a one-step proof (OSP). Ethereum can then verify this OSP by itself and, as the neutral referee, declare a winner.

The "rules" of the dispute involve parties making claims with proof to reach the single point of disagreement. Parties "narrow down" their claims via moves called bisections. After a party has made a bisection, there is nothing else left to do until another party comes in and counters it. The core of the system is that an honest party winning a one-step proof leaves the malicious party with no other moves to make. Once the honest party has accumulated enough time without being countered, it is declared the winner.

Compared to other dispute protocols, however, BoLD isnot a dispute between two specific Ethereum addresses, such as

Alice and Bob. Instead, it is a dispute between an absolute, correct history vs. an incorrect one. Claims in BoLD are not attached to a particular address or validator but instead to Merkle commitments of an Arbitrum chain's history. If Alice and Charlie are both honest, and Bob is malicious, Alice and Charlie can play the game as part of a single "team". If Alice goes offline in the middle of a dispute-game, Charlie can continue resolving the game on behalf of the honest team because Charlie and Alice claim and make moves on the correct history. This is why we say BoLD enables "trustless cooperation," as there is no need for communication between honest parties. We believe committing a set of chain history hashes instead of a specific hash at a moment in time is crucial for securing dispute protocols.

Spamming the dispute game

BoLD is a dispute-game in which the party that has accumulated seven days "not-countered" wins. That is, parties are incentivized to counter any new claims as soon as they appear to "block" their rivals from increasing their timers. For honest parties, responding to claims may sometimes require offchain computational work and, therefore, resources such as CPUs. However, malicious parties can make claims that are eventually found to be junk while making honest parties do actual work.

Because malicious parties can submit incorrect claims that make honest parties do work, there has to be an economic cost associated with making moves in the dispute-game. Said differently, we need a way to preventspam attacks in dispute games.

The cost of moves

When pricing the bonds required to make claims within disputes, we consider the marginal costs that the honest party incurs for each claim a malicious party makes. The BoLD research paper includes information such as the number of adversary moves multiplied by the gas cost of making bisections and claims and some estimates of the offchain computational costs. We deem this themarginal cost of a party in a dispute.

With BoLD, the space of disagreements between parties is of max size 2^69. As such, the dispute game has to be played at different levels of granularity to make it computationally feasible.

Let's use an analogy: say we have two one-meter sticks that seem identical, and we want to determine where they differ. They appear identical at the centimeter level, so we need to go down to the millimeter level, then the micrometer level, and then figure out where they differ at the nanometer level.

This is what BoLD does over the space of disputes. Parties play the same game at different levels of granularity. At the centimeter level, each centimeter could trigger a millimeter dispute, and each millimeter dispute could have many micrometer disputes, etc. This dispute pattern could be abused unless spam is discouraged.

Preventing spam

Since Ethereum knows nothing about which claims are honest or malicious until a one-step proof is provided, how can the protocol detect and discourage spam? A key insight is that honest parties only need to make one honest claim. Honest parties will never spam and create thousands of conflicting claims with themselves. Given this, we can put a price tag on making moves by looking at something called the "resource ratio" between honest and malicious parties, as defined in the BoLD research paper.

This ratio is computed as gas plus staking (or bonding) marginal costs of the adversary to the honest party. This means that certain values input into the equations can lead todifferent ratios . For instance, we can say the adversary has to pay10x the marginal costs of the honest party. However, aiming to increase this ratio significantly by plugging in different values leads to higher costs for all parties.

Dispute mini-bonds

We require parties to lock up some capital called a "mini-bond" when making big claims in a dispute. These bonds are not needed when making bisection moves but are critical for posting an initial claim. Pricing these mini-bonds helps achieve a high resource ratio of dishonest parties to honest parties.

It is clear that if we can multiply the cost to the malicious party by some multiplier of the honest party, we will get significant security benefits. For instance, imagine if a 1 billion dollar attack can be defended by simply pooling together 10 million dollars. Is it possible to achieve such a ratio?

Let's explore the limitations of making the cost to malicious parties higher than the honest parties'.

If we aim to have a constant resource ratio > 1, we have to do the following: if the adversary makes N stakes at any level, they can force the honest party to make N stakes at the next level down, where the adversary can choose not to place any stakes at all. Regarding resource ratio, to make the adversary always pay 10x in staking, we need to make the bond amount at one level 10x more than the next. As there are multiple levels, the equations for the bond size include an exponential factor on the desired constant resource ratio > 1.

Below, we plot the bond size vs. the resource ratio of malicious to honest costs. The source for these equations can be found in the research paper and isrepresented in this calculator .

If we desire a constant resource ratio of malicious to honest costs > 1, the required bond size inETH increases as a polynomial at a particular challenge level.

Trade-offs

Having a 1000x resource ratio would be nice in theory, but it would, unfortunately, require a bond of 1METH (2.56B USD at time of writing) to open a challenge in the first place, which is unreasonable. Instead, we can explore a more feasible ratio.

The resource ratio will drive the price of disputes claims, impacting both honest and malicious parties. However, claims canalways be made through atrustless pool . Honest parties can pool together funds to participate in disputes.

The sweet spot

So, now that we've established that a higher resource ratio is better but comes with some trade-offs, what is the sweet spot?

We propose a resource of ratio of 6.46 for Arbitrum One. While odd, this resource ratio was calculated taking the initial "bond" to become a proposer (mentioned earlier) and a worst case scenario of 500 gwei/gas on L1 for posting assertions and making sub-challenge moves (i.e. if an attack were to happen, the malicious actor could choose to perform their attack during a period of elevated gas prices). Again, we should emphasize that the ratio of malicious to honest cost should be high to sufficiently deter attacks. Under our current assumptions (500gwei/gas) and proposed parameters (bond sizes, etc) for every 6.46 spent by malicious parties attacking, only 1 is needed to defend it successfully in BoLD. Here's adirect link to the calculations where the X-axis is L1 gas costs in gwei and the Y-axis is the resource ratio.

Unfortunately, there is no "one size fits all" framework for choosing the resource ratio for your chain. Therefore, we recommend teams learn and understand the benefits and trade-offs of operating BoLD in a permissionless format - including performing the same type of economic risk analyses we have performed for Arbitrum One.

Thinking about incentives

Although we have made claims with hard numbers about how to price disputes and withdrawal delays in Arbitrum BoLD, we also took a step back and considered the theoretical assumptions we were making. Arbitrum One is a complex protocol used by many groups of people with different incentives. The research team atOffchain Labs has spent considerable effort studying the game theory of validators in Optimistic Rollup. Honest parties represent everyone with funds onchain, and they have a significant amount to gain by winning the challenge - as they can prevent the loss of their assets rather than losing them.

A more complex model is proposed, which considers all parties staking and their associated costs paper"Incentive Schemes for Rollup Validators" . The paper examines the incentives needed to get parties to check whether assertions are correct. It finds that there is no pure strategy, Nash equilibrium, and only a mixed equilibrium if there is no incentive for honest validators. However, the research showed a pure strategy equilibrium can be reached if honest parties are incentivized tocheck results. The problem of honest validators' "free riding" and not checking is well-documented as theverifier's dilemma . We believe future iterations of BOLD could include "attention challenges" that reward honest validators for also doing their job.

Service fee for “Active” proposers

For Arbitrum BoLD's initial launch, we believe that chain owners should pay a service fee to active, top-level proposers as a way of removing the disincentive for participation by honest parties who bond their own capital and propose assertions for Arbitrum One. The fee should be denominated inETH and should correlate to the annualized income that Ethereum mainnet validators receive, over the same time period. At the time of writing, the estimated annual income for Ethereum mainnet validators is approximately 3% to 4% of their stake (based onCoinDesk Indices Composite Ether Staking Rate (CESR) benchmark andRated.Network).

This service fee can be paid out upon an active proposer’s top-level assertion being confirmed on Ethereum and will be calculated using the duration of time that the proposer was considered active by the protocol. The procedure that calculates this will be handled off-chain, using a procedure that will be published at a later date. BoLD makes it permissionless for any validator to become a proposer and also introduces a way to pay a service fee to honest parties for locking up capital to do so. Validators are not considered active proposers until they successfully propose an assertion with a bond.

In order to become an active proposer for an Arbitrum chain, post-BoLD, a validator has to propose a state assertion to its parent chain. For Arbitrum One and Nova, the state assertion is posted onto L1 Ethereum. If they do not have an active bond on L1, they then need to attach a bond to their assertion in order to successfully post the assertion. Subsequent assertions posted by the same address will simply move the already-supplied bond to their latest proposed assertion. Meanwhile, if an entity, say Bob, has posted a successor assertion to one previously made by another entity, Alice, then Bob would be considered by the protocol to be the current active proposer. Alice would no longer be considered by the protocol as the active proposer and once Alice's assertion is confirmed, then Alice gets her assertion bond refunded. There can only be 1 "active" proposer at any point in time.

For Arbitrum One specifically, all eligible entities who wish to be paid this service fee from the Arbitrum Foundation must undergo the Arbitrum Foundation's KYC process as no AIP "may be in violation of applicable laws, in particular sanctions-related regulations". This is also written in the[ArbitrumDAO's Constitution](#) .

Rewards and Reimbursements for Defenders

The service fee described above is meant to incentivize or reimburse an honest, active proposer for locking up their capital to propose assertions and advance the chain. Similarly, in the event of an attack, a bounty is proposed to be paid out to honest defenders using confiscated funds from malicious actors (in the event of a challenge).

For Arbitrum One specifically, 1% (called the "defender's bounty") of the confiscated funds from a malicious actor is to be rewarded to honest parties who deposit a challenge bond and post assertions as part of a sub-challenge, proportional to the amount that a defender has put up to defend a correct state assertion during the challenge. This bounty applies for all challenges (block challenges, sub challenges and one step challenges). Note that any gas costs spent by honest parties to defend Arbitrum One during a challenge is 100% refundable by the Arbitrum Foundation. In this model, honest defenders and proposers of Arbitrum One are incentivized to participate while malicious actors stand to lose everything they spent attacking Arbitrum One. We believe chain owners who are interested in adopting BoLD for their own chain should follow a similar approach, described above for Arbitrum One, to incentivize challenge participation (but not necessarily assertion proposing).

In this design, defenders are only eligible for the defender's bounty if they deposit a challenge bond (for Arbitrum One, this is either 555 or 79ETH , depending on the level), posted to an on-chain assertion as part of a sub-challenge (i.e., not the top-level assertion), and have had their on-chain sub-challenge assertion get confirmed by the protocol. For Arbitrum One, the calculation for the defender's bounty is conducted off-chain by the Arbitrum Foundation, and payment will be made via an ArbitrumDAO governance vote (since confiscated funds go to an ArbitrumDAO-controlled address). Honest parties are not automatically rewarded with all the funds seized from malicious actors to avoid creating a situation where honest parties wastefully compete to be the first ones to make each honest move in the interactive fraud-proof game. Additionally, BoLD resolves disputes by determining which top-level assertion is correct, without necessarily being able to classify every move as "honest" or "malicious" as part of the interactive fraud-proof game without off-chain knowledge.

Once all of a validator's proposed assertions are confirmed, a validator can withdraw their bond in full. Additionally, the protocol will automatically handle refunds of challenge bonds for honest parties and confiscation of bonds from malicious parties in the event of a challenge. In other words, bonds put up by honest parties will always be returned and bonds put up by malicious parties will always be confiscated. For Arbitrum One specifically, L1 gas costs for honest parties defending a challenge will be reimbursed by the Arbitrum Foundation using a procedure that will be published at a later date. The chain owner could therefore consider the cost of incentivizing or lending the assets to a single honest proposer in the happy case as thesecurity budget of the chain .

For Arbitrum One specifically, all eligible entities who wish to be paid the defender's bounty from the ArbitrumDAO must undergo the Arbitrum Foundation's KYC process as no AIP "may be in violation of applicable laws, in particular sanctions-related regulations". This is also written in the[ArbitrumDAO's Constitution](#) .

Conclusion

This page summarizes the rationale behind choosing bond sizes and the cost of spam prevention in Optimistic Rollup dispute protocols. We recommend that bond sizes be high enough to discourage challenges from being opened, as malicious parties will always stand to lose when playing the game. As Arbitrum BoLD does not tie disputes to specific addresses, honest parties can have trustless cooperation to resolve disputes if desired. We posit that making the cost of the malicious parties 10x that of the honest party leads to nice economic properties that help us reason about how to price bonds. Finally, we look at a high-level game theory discussion of Optimistic Rollups and argue that solving the verifier's dilemma by incentives to honest validators is an important addition to work towards.

The topic of further improvements and new economic and incentive models for BoLD are valuable and we believe it deserves the full focus and attention of the community in future proposals and discussions. Details around additional or new proposed economic or incentive models for BoLD will need continued research and development work, but the deployment of BoLD as-is represents a substantial improvement to the security of Arbitrum even without economic-related concerns resolved. Edit this page Last updatedonJan 27, 2025 Previous Technical deep dive Next A gentle introduction