Thanks to @adlerjohn for comments on this.

Problem: layer two fraud proofs can be censored

In the traditional fraud proofs model with light clients, censorship by block producers is not possible, because fraud proofs are distributed across the peer-to-peer networking layer, not posted on-chain. Furthermore, light clients reject invalid blocks regardless of when the fraud proof was received, as there is no dispute period. If a main chain block is invalid, it's invalid.

In the Ethereum flavour of optimistic rollups and other layer two models, fraud proofs are processed by a consensus-critical execution layer on the main chain, and there's a dispute period for sidechain block headers. This is because if we allow a fraud proof after the dispute period, there is no mechanism to roll back all the state that it has impacted on the main chain.

The problem is that these fraud proofs can be censored by the consensus group of the main chain by refusing to include them before the dispute period, thus causing a safety failure in layer two systems. This is bad® because normally, censorship means a liveness failure, not a safety failure.

A solution: censorship resistance via weak subjectivity

We can, however, allow for censorship-resistant fraud proofs for Ethereum flavour sidechains by also distributing the fraud proofs in the peer-to-peer network layer, and using a weak subjectivity assumption.

To achieve this, we must add new fields to blocks called "extension fields". Each transaction gets its own dedicated extension field. These extension fields are not committed to in the block header. This also means transactions with empty extension fields do not take up additional block space. (This means these fields aren't really a part of the block at all, but this is the way I logically think about it, as fields are attached to txes at specific blocks.)

An extension field can be in one of two states: \mathsf{null}

, or have some value e

such that it returns true on some predicate p(\mathsf{tx}, e) = {\mathsf{true}, \mathsf{false}}

, where p

is defined by the contract that \mathsf{tx}

calls. Thus, an extension field can only be filled a value other than \mathsf{null}

if there is some valid e

that causes p(\mathsf{tx}, e)

to be \mathsf{true}

. In the case of optimistic rollups, \mathsf{tx}

can be a transaction by an aggregator posting a new sidechain block header, and e

is a fraud proof. Thus, p

can be a predicate that returns \mathsf{true}

if and only if e

is a valid fraud proof for the block header in \mathsf{tx}

.

When a fraud proof for an optimistic rollup sidechain is generated (or more generally, any value e

that satisfies a contract's predicate), it is sent to all nodes on the main chain via the peer-to-peer network layer, who only relay it if the fraud proof is correct according to the contract-defined predicate. The nodes then update the extension field value for that transaction from \mathsf{null}

to e

. All the nodes who receive it within a contract-defined dispute period minus a conservative maximum network delay will automatically soft fork, by rejecting any block posted one block ID before the end of the dispute that does not post this fraud proof (or more generally, any e

that satisfies the predicate) on-chain, so that the contract can process it.

If the fraud proof is generated after the dispute period, then new nodes will not know if it was generated before or after the dispute period, so they cannot determine if they should join the soft forks. Thus, a weak subjectivity assumption is required for bootstrapping new nodes.

Gas fees

An open question is how to price gas fees for executing predicates.

To prevent arbitrarily expensive-to-compute predicates from spamming the chain at no cost, each \mathsf{tx}

must pre-pay for the gas required to evaluate the predicate at a certain upper-bound computational cost decided by the contract, and gets refunded if no valid values that satisfy the predicate are sent within the dispute period. The contract defines the gas amount that must be pre-payed, and for an optimistic rollup, the sidechain considers sidechain transactions that use more gas than the pre-pay amount (minus a bit) to be invalid, which ought to be provable in a fraud proof executed by the main chain by only executing the invalid transaction up to a certain gas usage.

Although the pre-pay gas amount is chosen by the contract, the wei/gas is decided by the transaction submitter, participating in a fee market. The transaction submitter might choose 0 wei/gas, and if a block producer mines this, they can force a future block producer to evaluate the predicate and include a fraud proof in the future for free, or risk their blocks being rejected due to censorship. To prevent this, we could force the wei/gas paid to be higher than the median wei/gas paid in the past 1000 blocks. We then hope that by the time the dispute period ends, the wei/gas wouldn’t have fluctuated so much that a block producer would be forced to mine a not-as-profitable transaction, or the transaction submitter paid too much.