I'd love some help thinking through a potential construction.

The core idea is, what if we want to do ZK rollups, but only post a single commitment on chain per block, as in Plasma?

It would work as follows: there is a bonded operator for the chain, who can post Merkle roots on chain. The data they Merkelize is a ZK rollup block, containing a set of input UTXOs + deposits, a set of output UTXOs + exits, and a proof that the all signatures are valid and the transition is valid (basically, a standard ZK rollup block). The commitment has a timeout period, similar to optimistic rollup, during which validity and data availability challenges can revert the block. In case of reversion, the last commitment before it is returned to (from some sort of accumulator, I suppose).

If the block is available, but invalid (invalid UTXOs, invalid validity proof), anyone can put up a bond to challenge the operator on the main chain. In order to win the challenge, the operator must post the block on the main chain, and it must pass a validity check just as in a regular ZK rollup contract, and must have a UTXO set that is identical to the previous UTXO set + new deposits - exits. They won't be able to submit such a transaction if the block is invalid.

If the block is unavailable, someone needs to post an availability challenge to the operator. The operator can answer it by posting the block data on chain. This remains a potential griefing vector. (EDIT: actually, anyone can respond to the availability challenge).

In order to withdraw, participants must enter an exit queue, where they submit a UTXO they want to exit and a Merkle proof that the transaction was included as an exit in a recent block. So long as the UTXO is valid, and the block they exited does not get reverted within the timeout, their exit will go through.

Surely there has to be something wrong with this scheme, right? Besides the griefing vector. I think this is somewhat similar to Plasma is Plasma.