Plasma Vector

Nathan Ginnever

December 14, 2018

Introduction

This work presents an application of cryptographic universal accumulators [5] that was expanded upon in the blockchain setting by Bunnz et al [1]. We apply these accumulators in Plasma [2] and show that with minimal "roll back" in the case of data unavaibility, Plasma Vector can scale to high transaction throughput with low overhead for clients at the expensive of increased prover complexity from the operator or service nodes. Vectors [] (VC) have the property of binding data to a position in the list and we explore why this is a desired property of accumulators in the blockchain application setting. Our contribution is to reducing the inclusion and exlcusion proof size of Plasma Cash/flow [] from an $O(n)$ and $O(logn*b)$ respectively, where b is a number of blocks commited and n is the number of coins a user owns, to an $O(1)$ for both. We will see that replacing the merkle tree entirly with a VC accumulator can increase the granularity of Plasma Cash/flow to allow for smaller payments.

As with previous Plasma designs, this paper will detail the *deposit*, *send*, and *exit* mechanisms of the UTXO set based transaction system applied to specific indexes. This expands on the Plasma MVP construction, reducing the verification overhead and removing the mass exit problem.