Nitro protocol is completely specified protocol for building generalized, n-party state channel networks.

The high-level overview of the approach is as follows:

1. We set up an on-chain system of balances. Balances can correspond to both participants and channels.
2. We view a state channel as a device that allows a fixed set of participants to reach an outcome by executing a multi-party application.
3. The outcome can be used to update the balances on-chain, including allowing one channel to pay out to another.

The strength comes from the fact that, armed with the knowledge that the outcome could

update the balances on-chain if required, it becomes possible for the parties to do the majority of the rebalancing off-chain.

This approach decouples the operation of the state channels from the network that they are part of, allowing the state channels to update independently and making it easier to reason about their behavior. We don't require atomic updates across multiple channels (e.g. the Counterfactual root nonce) nor do channels acquire additional constraints from being part of at the network (e.g. the validity time limit seen in the Perun approach).

In the paper we present two protocols:

- Turbo Protocol: enables ledger channels, allowing multiple channels between the same set of participants to be supported by a single on-chain deposit.
- Nitro Protocol: enables virtual channels, allowing a set of participants who share on-chain deposits with a common intermediary to open a channel without a shared on-chain deposit of their own.

The bulk of the paper is spend presenting constructions/procedures in these two protocols, including partial withdrawals and top-ups for ledger channels and an example of a 3-party virtual channel. We also provide algorithms for calculating the value attributable to each channel/participant in the network and develop the theory for reasoning about the correctness of the protocol.

We have a v1 implementation of the protocol in solidity and are currently working on client/server code to allow us to run a state channel application on top of it.