Chapter 5. Recover from Consensus Failures

▪ Detecting and Rejecting Invalid Transactions

Consensus algorithm of Locus Chain is based on an idea of verifying transactions properly propagated to certain nodes in a certain time. We assume that transactions are issued properly by honest accounts and nodes, but there could exist some 'invalid' transactions, accidentally or not.

'Invalid transaction' is a transaction in which the containing information contradicts with the existing confirmed transactions and other information. For example, a transaction with a sign that cannot be verified with known account public key is an invalid transaction. Another example is that a transaction with the transaction index not following existing previous transactions is also an invalid transaction.

Every node in a shard must validate received transactions to detect invalid transactions. Such invalid transactions are ignored by nodes to prevent further propagation.

There are cases where a newly received transaction does not contradict with the previously confirmed transactions, but contradicts with other transactions in the same round. For example, if an account issues two transactions with the same transaction index, each transaction may not contradict with the previously confirmed rounds, but the two transactions contradict each other. If these contradicting transactions are separately sent to different nodes, the receiving nodes cannot reject the received transaction because the transactions does not violate previous rounds individually. In such a situation, the whole shard may contain contradicting transactions, which is a problematic 'double-spending' situation. However, nodes may resolve conflicts by some predefined rules when both transactions are received, usually before the end of the round.

▪ Cases of Consensus Failure

If most of the nodes are honest and the network connection is stable, it is quite unlikely to fail to reach a consensus. However, failure can occur in the following corners.

- ✓ There was no Round State Proposal in Step 2:
  In this case, a timeout will occur in step 2 and step 3. Consequently, step 4 will reach an agreement of "consensus failure" with more than 2/3 votes of "Confirmation Failure Messages."

- ✓ No Round State Selection Vote gained over 2/3 votes in Step 3:

In this case, a timeout will occur in step 3. Consequently, step 4 will reach an agreement of “consensus failure” with more than 2/3 votes of “Confirmation Failure Messages.”

- ✓ No Round State Confirmation or Confirmation Failure Vote gained over 2/3 votes in Step 4:
  In this case, a timeout will occur in step 4, leaving consensus result “undecided.” This is the most problematic case. A critical network failure may prevent proper communication in a shard, and there could be inconsistencies of consensus result. In other words, some major number of nodes may not receive the proper result of consensus at the end of the round, leaving the round undecided for the major number of nodes.

- **Retrying consensus on failure**

When a Round Consensus is reached by major votes of Confirmation Failure Messages on Step 4, or consensus is undecided by a timeout of Step 4, each node regards the round as ‘failed’ and should start a re-agreement process, or process of retrying a Round Consensus.

The consensus algorithm for retrying a Round Consensus is the same with the normal consensus algorithm of Locus Chain, but with parameters changed. All transactions of failed rounds will be included in the re-agreement process. (Transactions would be more perfectly propagated over the shard because a certain time has passed from the end of the round.) Each node may re-sync received transactions with other nodes.

The number of committee nodes for the re-agreement process is increased from the normal process. If two or more confirmed round states are discovered later, the total stake of committee nodes could be used as a good criterion for pruning unsuitable states. If a node may see multiple different confirmed round states, the one with the largest total stake must be selected. (This special case may occur when a round confirmation result was not propagated for some network failure, and later re-broadcasted after network recovery.)

In the next chapter, the rewards for node activities (aka “mining) are explained, including Coin and Gas values.