Double Spending Problem

Jirayu Petchhan, D10907801

1. **Centralized Trust**

Problem = Untrustless

Commit fraud (cheat) = able to hide and/or steal some amount in transaction list

1. **Multiple P2P Centralized Trust**

Better than 1.? = Much better with 2 or more ledger

Problem = more Latency during cross-checking for corresponding transaction lists

How synchronize? = cross-checking between two ledgers by making the recordings from both places in sync

Commit fraud = If there are only 2 ledgers, there might be one ledger has recorded with fake transaction how can we separate which one is correct?

But if there are N ledgers that can make a trustless system.

1. **Centralized Trust with Immutable Ledger**

Problem = Separation of block types by ownership

Commit fraud = Ledger may send blogs with the same sum to others.

1. **Multiple Centralized Trust with Immutable Ledger with Blocks**

Problem = Latency of cross-checking between two ledgers and, in case of transaction blocks have equally, how do we separate it? Who's own this box? Because we only know the net amount, the sum of the blocks.

Commit fraud = If there are only 2 ledgers, there might be one ledger has recorded with fake transaction how can we separate which one is correct? The same cheating as Multiple P2P case.

But if there are N ledgers that can make a trustless system.

1. **Multiple Decentralized Trust with Immutable Ledger**

Solve = every block has to check altogether, can't add or subtract amount (immutability) in each block until there are transaction required and every block have to cooperate checking.

Other issues = physical issues (e.g. privacy)

Function = How to separate the type(address) of each block

Trustless? = yes, every block has to check transaction altogether.

Double spending check? = yes, because every block has all blocks connected to chain that cannot be deleted and edited manually. Until the transaction comes in.

Commit Fraud = Block attacked from physical and/or cyber-security attacking