Incentives in computer science
Blockchain

- ## Bitcoin

Bitcoin is a decentralized digital currency, without a central bank or single administrator, that can be sent from user to user on the peer-to-peer bitcoin network without the need for intermediaries.
So how does Bitcoin work? The basic primitive is that of a transaction, which includes the following ingredients.

A Bitcoin Transaction

1. One or more senders.
2. One or more receivers.
3. The amount of BTC (Bitcoins) transferred from each sender to each receiver.
4. A proof of ownership of the coins being transferred, in the form of a pointer back to most recent transactions involving the transferred coins.
5. A transaction fee, paid by the sender to the authorizer of the transaction.

A transaction is valid if:

1. It has been cryptographically signed by all of the senders. (This can be verified using the senders' public keys.)
2. The senders really do own the coins being transferred.

The second criterion is also easy to verify, given how Bitcoin works. Specifically, transactions are broadcast to all other users (through a peer-to-peer network), and all users keep track of all transactions that have ever been authorized. Thus, everyone knows everyone's current balance, and whether they're in a position to make the specified transfer. The record of all transactions that have been authorized so far is called the *ledger*, and it is the sun around which the Bitcoin world orbits.

● Bitcoin's Blockchain Protocol

➔ Blocks

Transactions are added to the ledger in groups (rather than one-by-one), known as blocks. Specifically, a block has the following ingredients:

1. One or more transactions.
2. A hash of the previous block.
3. A nonce. (I.e., a bunch of bits that can be set arbitrarily.)

➔ The High-Level Idea

1. Any user can authorize a block. Bitcoin incentivizes users to do authorizations through explicit monetary rewards (in BTC, naturally).
2. To avoid anarchy, authorizing a new block of transactions involves a proof of work, meaning that the authorizer has to solve a computationally difficult puzzle.

➔ Computationally Difficult Puzzles

An important definition: a block b is valid if h(b) is sufficiently close to 0, where h is a pre-agreed upon hash function (currently, SHA-256, so h(b) is 256 bits). Thus finding a valid block more or less involves inverting a one-way/cryptographic hash function. By "sufficiently close to 0," we mean that the leading l bits of h(b) should all be 0, where l is a parameter. Note that l provides a knob to control the difficulty of the problem: the bigger the choice of l, the harder the problem.

➔ Block Rewards and Bitcoin Mining

Bitcoin mining is the processing of transactions in the digital currency system, in which the records of current Bitcoin transactions, known as blocks, are added to the record of past transactions, known as the block chain.The intended behavior for a bitcoin miner is: choose a subset of the outstanding transactions that it knows about, insert the hash of the current last block of the blockchain, arbitrarily set the bits in the nonce, and hope that the resulting block b is valid.

Block rewards are new bitcoins awarded to cryptocurrency miners for being the first to solve a complex math problem and creating a new block of verified bitcoin transactions.

➔ Forks

Once in a while, two different bitcoin miners will discover valid blocks at roughly the same time. This results in a fork in the blockchain, where two valid blocks, each with its own set of transactions, point to the same previous block. Forks are related to the fact that different parties need to use common rules to maintain the history of the blockchain. When parties are not in agreement, alternative chains may emerge. While most forks are short-lived some are permanent. There needs to be a mechanism for deciding which branch of the fork is the “right” one, for two reasons:

(i) so that everybody knows which transactions have been authorized; and

(ii) so that bitcoin miners know which block they should be trying to extend.

The Bitcoin protocol specifies the intended behavior when there’s a fork: a user should regard the longest branch as the valid one, breaking ties according to the block that it heard about first. When there is a fork as, it is completely possible that different users will have different opinions about which branch is the valid one.

Eventually, some bitcoin miner will authorize a new block, which extends only one of the branches (depending on which hash the miner put in the new block). If there were previously branches with equal length, then this new block will break the tie and create a chain longer than any other. At this point, all users will again have a consistent view of the blockchain (as

the longest chain). When this happens, blocks not on this longest chain are "orphaned," and the transactions in them are not regarded as authorized.

➔ Sybil Attacks

Bitcoin users are identified with public keys. It's not hard to generate many public keys, so many Bitcoin "users" might actually correspond to the same person. Deliberately creating multiple identities in a system is often called a Sybil attack. 9 Sybil attacks plague many systems (see the next lecture), but remarkably, they cause no issues in Bitcoin. Your influence in Bitcoin is determined entirely by the amount of computational power that you wield—the number of identities is irrelevant.

• Incentive issues

➔ The double spend attack

The first type of deviation we'll look at is when miners deliberately create forks. This is simple to do: when searching for a valid block, just insert the hash of a block on the blockchain that is not the last block.

To see why a miner might want to create a fork, suppose in the transaction T, Alice transfers some bitcoins to Bob. Suppose this transaction gets added to the blockchain as part of block b1. Per the discussion above, Bob only ships the purchased goods to Alice once another block b2 has been appended to b1. When Alice has the goods, she could try the following attack: try to find a valid block b3 extending b0, another block b4 extending b3, and a third block b5 extending b4. Alice does not put transaction T in any of these blocks. If Alice successfully creates these three blocks before any other miner extends b2, then she rips off Bob: b1 and b2 are orphaned and Alice's payment to Bob gets canceled, while the goods have already

been sent.10 This attack is sometimes called the double-spend attack, especially in the case where Alice puts a payment T 0 to Carol in the block b3, promising the same coins to Carol that she already promised to Bob. Since bits are easily copied, every digital currency must address double-spending attacks.

➔ The 51% Attack

Bitcoin is not intended to function when a single entity controls more than 50% of the computational power. Such an entity can effectively act as a centralized authority, defeating the whole point of Bitcoin. For example, while such an entity cannot outright steal bitcoins from another user's account (because of the cryptographic protections), it can freeze the assets of any user that it wants, by forcing any blocks involving that user to be orphaned. In general, it is only interesting to study Bitcoin when no one controls more than 50% of the overall computational power.

➔ Selfish Mining

Selfish mining is a strategy for mining bitcoin or other cryptocurrencies in which groups of miners collude to increase their revenue and exert power over a blockchain. "Mining" is the process by which nodes in the blockchain's network validate and confirm transactions, with miners earning newly minted tokens in return for their computational effort. With selfish mining, the cartel obscures newly created blocks from the main chain, revealing them at a later point in time.
By selfish mining, a miner (or group of miners) increases their revenue by strategically withholding and releasing blocks to the network. Typically, we expect a miner to announce a block as soon as they find it. If the block is confirmed, they will get the block reward.

By not broadcasting their block right away, though, these miners effectively create their own private branch of the blockchain. The rest of the network continues to build on the previous block, while the selfish miner builds on

top of this new chain. From that point, both chains will look completely different.

The goal of the selfish miner is to always remain at least one block ahead of the rest of the network. Nodes accept the chain with the most accumulated proof of work as the valid blockchain. At any time, the selfish miner can reveal their chain. If it is longer than the one followed by the rest of the network, the existing blocks will be discarded, and transactions reversed. The miner collects all of the rewards from these blocks and causes other parties to waste resources.