Handout 7 (Bitcoins)

In my opinion Bitcoins are an elaborate Ponzi scheme[1]—still the ideas behind them are really beautiful and not too difficult to understand. Since many colourful claims about Bitcoins float around in the mainstream media, it will be instructive to re-examine such claims from a more technically informed vantage point. For example, it is often claimed that Bitcoins are anonymous and free from any potential government meddling. It turns out that the first claim ignores a lot of research in de-anonymising social networks, and the second underestimates the persuasive means a government has at their disposal.

There are a lot of articles, blogposts and so on available about Bitcoins. Below I will follow closely the very readable explanations from

http://www.michaelnielsen.org/ddi/how-the-bitcoin-protocol-actually-works/ and

http://www.imponderablethings.com/2013/07/how-bitcoin-works-under-hood.html

The latter also contains a link to a nice youtube video.

Let us start with the question who invented Bitcoins? You could not make up the answer, but we actually do not know who the inventor is. All we know is that the first paper

https://bitcoin.org/bitcoin.pdf

is signed by Satoshi Nakamoto, which however is likely only a pen name. There is a lot of speculation who could be the inventor, or inventors, but we simply do not know. This part of Bitcoins is definitely anonymous. The first Bitcoin transaction was made in January 2009. The rules in Bitcoin are set up so that there will ever only be 21 Million Bitcoins with the maximum reached around the year 2140. Contrast this with traditional fiat currencies where money can be printed almost at will. The smallest unit of a Bitcoin is called a Satoshi which is the 10^{-8}th part of a Bitcoin. Remember a Penny is the 10^{-2}th part of a Pound.

The two main cryptographic building blocks of Bitcoins are cryptographic hashing (SHA-256) and public-private keys using the elliptic-curve encryption scheme for digital signatures. Hashes are used to generate 'fingerprints' of data that ensure integrity. Public-private keys are used for signatures. For example sending a message, say *msg*, together with the encrypted version

$$msg, \{msg\}_{K^{priv}}$$

allows everybody with access to the public key to verify the message came from the person who knew the private key. Signatures are used in Bitcoins for verifying the addresses where the Bitcoins are sent from. Addresses in Bitcoins are essentially the public keys. There are 2^{160} possible addresses, which is such a

[1] http://en.wikipedia.org/wiki/Ponzi_scheme

vast amount that there is not even a check for duplicates, or already used addresses. If you start with a random number to generate a public-private key pair it is very unlikely that you step on somebody else's shoes. Compare this with email-addresses you ever wanted to register with, say, Googlemail, but which were always already taken.

One main difference between Bitcoins and, say, traditional banking is that you do not have a place that records the balance on your account. Traditional banking involves a central ledger which specifies the current balance in each account, for example

account	balance
Alice	£10.01
Bob	£4.99
Charlie	-£1.23
Eve	£0.00

Bitcoins work differently in that there is no such central ledger, but instead a public record of all transactions ever made. This means spending money corresponds to sending messages of the (rough) form

$$\{\text{I, Alice, am giving Bob one Bitcoin.}\}_{K^{priv}_{Alice}} \tag{1}$$

These are the transactions that are the only data that is ever stored (we will come to the precise details later on). The transactions are encrypted with Alice's private key such that everybody, including Bob, can use Alice's public key K^{pub}_{Alice} for verifying that this message came really from Alice, or more precisely from the person who knows K^{priv}_{Alice}.

The problem with such messages in a distributed system is what happens if Bob receives 10, say, of these transactions. Did Alice intend to send him 10 Bitcoins, or did the message get duplicated by for example an attacker re-playing a sniffed message? What is needed is a kind of serial number for such transactions. This means transaction messages look more like

$$\{\text{I, Alice, am giving Bob Bitcoin \#1234567.}\}_{K^{priv}_{Alice}}$$

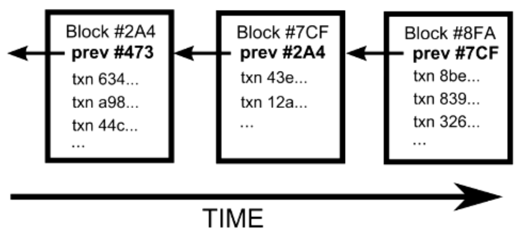
There are two difficulties, however, that need to be solved. One is who is assigning serial numbers to Bitcoins and also how can Bob verify that Alice actually owns this Bitcoin to pay him? In a system with a bank as trusted third-party, Bob could do the following:

- Bob asks the bank whether the Bitcoin with that serial number belongs to Alice and Alice hasn't already spent this Bitcoin.

- If yes, then Bob tells the bank he accepts this Bitcoin. The bank updates the records to show that the Bitcoin with that serial number is now in Bob's possession and no longer belongs to Alice.

But for this banks would need to be trusted and would also be an easy target for any government interference, for example. Think of the early days of music sharing where the company Napster was the single point of "failure" which was taken offline by law enforcement.

Bitcoin solves the problem of not wanting to rely on a bank by making everybody the "bank". Everybody who cares can have the entire transactions history starting with the first transaction made in January 2009. This history of transactions is called *blockchain*. Bob, for example, can use his copy of the blockchain for determining whether Alice owned the Bitcoin and if yes transmits the message to every other participant on the Bitcoin network. The blockchain looks roughly like a very long chain of individual blocks



Each block contains a list of individual transactions, called txn in the picture above, and also a reference to the previous block, called prev. The data in a block (txn's and prev) is hashed so that the reference and transactions in them cannot be tampered with. This hash is the unique serial number of each block. Since this previous-block-reference is also part of the hash, the whole chain is robust against tampering. I let you think why this is the case. …But does it eliminate all possibilities of fraud?

We can check the consistency of the blockchain by checking the entire blockchain whether the references and hashes are correctly recorded. I have not tried it myself, but it is said that with the current amount of data (appr. 12GB) it takes roughly a day to check the consistency of the blockchain on a "normal" computer. Fortunately this "extended" consistency check usually only needs to be done once.

Recall I wrote earlier that Bitcoins do not maintain a ledger listing all the current balances in each account. Instead they record only transactions. Therefore it is possible to extract from the blockchain a transaction graph that looks like the picture shown in Figure 1. Take for example the rightmost lower transaction from Charles to Emily. This transaction has as receiver the address of Emily and as the sender the address of Charles. In this way no Bitcoins can appear out of thin air (we will discuss later how Bitcoins are actually generated). If Charles did not have a transaction of at least the amount he wants to give Emily to his name (i.e. send to an address with his public-private key) then there is no way he can make a payment to Emily. Equally, if now Emily wants to pay for a coffee, say, with the Bitcoin she received from Charles she can only make a transaction to forward the message she received. The only slight complication

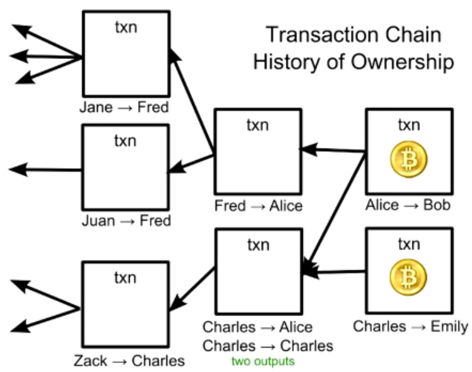


Figure 1: Transaction graph that is implicitly recorded in the public blockchain.

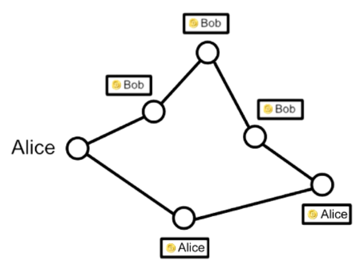
with is that incoming transactions can be combined in a transaction and "outgoing" Bitcoins can be split. For example in the leftmost upper transactions in Figure 1 Fred makes a payment to Alice. But this payment (or transaction) combines the Bitcoins that were send by Jane to Fred and also by Juan to Fred. This allows you to "consolidate" your funds: if there was always only a way to split transactions, then the amounts would get smaller and smaller. But it is also important to be able to split the money from one or more incoming transaction to more than one receiver. Consider again the rightmost transactions in Figure 1 and suppose Alice is a coffeeshop owner selling coffees for 1 Bitcoin. Charles received a transaction from Zack over 5 Bitcoins. How does he pay for the coffee? There is no real notion of change in the Bitcoin system. What Charles has to do instead is to make one single transaction with 1 Bitcoin to Alice and with 4 Bitcoins going back to himself. Which Charles can then use to give to Emily.

Let us make another example. Let us assume Emily received 4 Bitcoins from Charles and independently has another transaction (not shown in the picture) that sends 6 Bitcoins to her. If she now wants to buy a coffee from Alice for 1 Bitcoin she has two possibilities. She could just forward the transaction from Charles over 4 Bitcoins to Alice splitted in such a way that Alice receives 1 Bitcoin and Emily sends the remaining 3 Bitcoins 'back' to herself. In this case she would now be in the "posession" of two unspend Bitcoin transactions, one over 3 Bitcoins and the independent one over 6 Bitcoins. Or, Emily could combine both transactions (one over 4 Bitcoins from Charles and the independent one over 6 Bitcoins) and then split this amount with 1 Bitcoin going to Alice and 9 Bitcoins going back to herself.

I let you have time to let this concept of transactions sink in…in the words of a famous 60ies Band: "All you need is transactions". There is no need for a central ledger and no need for an account balance from traditional banking. The closest what Bitcoin has to offer for a notion of a balance in a bank account are the unspend transaction that a person (that is public-private key address) received. That means transactions that can still be forwarded.

Also consider the fact that whatever transaction is recorded in the blockchain is what will set the "historical record". If a transaction says 1 Bitcoin from address A to address B, then this is what will be recorded. This is also how Bitcoins can actually get lost: if you forget your private key and you had just a message forwarded to you address of the public key, then bad luck: you will never be able to forward this message again, because you will not be able to form a valid message that sends this to somebody else (we will see the details of this later). But this is also a way how you can get robbed of your Bitcoins. An attacker might get hold of your private key and then quickly forward the Bitcoins in your name to an address the attacker controls. You have never again access to these Bitcoins, because for the Bitcoin system they are assumed to be spend. And remember with Bitcoins you cannot appeal to any higher authority. Once it is gone, it is gone. This is different with traditional banking where at least you can try to harass the bank to rollback the transaction.

This brings us to back to problem of double spend. Say Bob is a merchant. How can he make sure that Alice does not cheat. She could for example send a transaction to Bob. But also to Charlie, or even herself. If Bob is also in the coffee business, he would potentially be cheated out of his money. The problem is how should people update their blockchain? You might end up with a picture like this



Alice convinced some part of the "world" that she is still owner of the bitcoin and some other part of the "world" thinks its Bob's.