Bitcoin

Giuseppe Di Battista and Roberto Tamassia
Bitcoin: a New Currency

• Digital
  • Stored and exchanged only in digital form
• Social
  • Maintained by an open network of people
  • Provides incentives for cooperative behavior
• Independent
  • Not associated with any government
  • Not guaranteed by any central bank
Bitcoin: a New Currency

- Transparent
  - Defined by open standards
  - All transactions recorded in a public ledger
- Finite
  - Fixed upper bound on the total number of bitcoins
- Cryptographic
  - Trust based on widely accepted cryptographic methods
Bitcoin: a New Currency

- Global
  - Can be acquired, spent, and transferred anywhere on the internet
- Widely accepted
  - Can be used to buy goods and services
  - Can be exchanged with traditional currencies
- Controversial
  - Economic, legal, and technical challenges
History of Bitcoin

• 2009: Satoshi Nakamoto’s paper
• 2009-2011: The early years
  • Underground
  • Community of enthusiasts
  • Price < $1
• 2012-: Growth
  • Wild price fluctuations, up to $1K
  • Media coverage
  • Increasing merchant acceptance
  • Government regulation attempts
  • Venture capital investment
Price of Bitcoin

Data source: blockchain.info
History of Bitcoin - Notable Events

• Price milestones
  • Dropped below $200 in early 2015

• Hacking and frauds
  • MtGox (2014) - hack or fraud?
  • Bitstamp (2015) hack and recovery

• Government intervention and regulation
  • Russia bans bitcoin (2014)
  • State of NY proposes bitcoin licenses (2014)
  • Australia considers VAT on bitcoin sales (2014)
  • US deems bitcoin property for tax purposes (2014)
  • Bank of Italy warns about risks of bitcoin (2015)
Bitcoin - Technical Introduction

- Addresses and transactions
- Blockchain
- Proof of work
- Consensus and forks
Bitcoin Address

• Bitcoins are generally “stored” at addresses
  • E.g. an address may contain 3.14159 bitcoins

• An address is a string of bits of at most 35 alphanumeric characters

Example (Apache Software Foundation):

1BtjAzWGLyAavUkbw3QsyzzNDKdtPXk95D
Bitcoin Transactions

- A typical bitcoin transaction transfers bitcoins from a set of input addresses storing them into another set of output addresses
  - Many-to-many transaction, different from traditional one-to-one wire transfers and credit card payments

- The bitcoins stored at the set of input addresses are transferred to the output addresses, arbitrarily redistributing among them
  - Example of a transaction redistributing 300.7 bitcoins:
    - Inputs: address1 (100.5 bitcoins) address2 (200.2 bitcoins)
    - Outputs: address3 (50.1 bitcoins), address4 (220 bitcoins), address5 (30.6 bitcoins)

- A given bitcoin address may appear as input or output address in multiple transactions
Questions

• Who owns the bitcoins at an address and can issue transactions from it?
  • Enabled by a public-key cryptosystem

• Who can view addresses and transactions?
  • Everyone – public ledger of all transactions

• Who can verify transactions?
  • Everyone – public-key cryptosystem

• How are bitcoin created to start with?
  • Initial creation – genesis
  • Incremental addition – mining
Bitcoin Addresses and Keys

- To enable ownership of addresses, bitcoin uses cryptographic hashing and a public-key cryptosystem.
- To create a new address, the owner creates a private-public key pair (SK, PK).
- The address A is then obtained by applying a cryptographic hash function h to PK, followed by an alphanumeric encoding e:
  \[ A = e(h(PK)) \]
- Encoding function e, hash function h, and the public-key cryptosystem are publicly known parameters of the bitcoin system.
  - Base58, SHA256, RIPEMD160, ECDSA
Proving Ownership of an Address

- Alice asks Bob to prove ownership of address A
- They mutually agree on a message M
- Bob gives Alice a proof consisting of
  - Public key PK
  - Digital signature S of M
    - S is obtained by applying the decryption algorithm to M with private key SK
Verifying an Address Proof of Ownership

- Bob has given Alice a proof of ownership of address A consisting of
  - Public key PK
  - Digital signature S of M
- Alice verifies the proof as follows
  - She checks that $A = e(h(PK))$
    - i.e., PK is the public key associated address A
  - She checks that $S$ is the signature of M
    - she applies the encryption algorithm to S using public key PK and checks that the the output is M
    - i.e., Bob knows the private key associated with PK
A Simple Transaction

- Alice wants to pay 3.14 bitcoins to Bob
- Alice owns 3.14 bitcoins at address A
- Bob tells Alice to send him the bitcoins at his address B
- To pay Bob, Alice creates a transaction essentially consisting of the following components (details to follow)
  - Addresses A and B
  - The public key associated with A
  - Amount 3.14
  - A digital signature over the message consisting of the above items, created with the private key associated with A
A Simple Transaction

- Only Alice can spend bitcoins at address A
  - nobody else has the private key for A
  - hence nobody else can create the digital signature required for a transaction with input A
- Everybody can inspect the transaction and verify the digital signature to determine the willingness of the owner of A to transfer 3.14 bitcoins to the owner of B
- The identity of Alice and Bob is not revealed by the transaction
Blockchain

- All bitcoin transactions ever issued are posted on a public ledger, called blockchain, that
  - groups transactions into blocks
  - linearly orders blocks
- Each block is referred to by an integer denoting its distance from the start block
- On average, a new block appears every 10 minutes and the number of transactions in a block varies
- Example: on January 19, 2015 12:44pm EST
  - the latest block is 339,673, which includes 2,144 transactions totaling 21,327.46 bitcoins transferred
Goals of the Blockchain

• Instill confidence in the use of bitcoin by preventing double spending, i.e., same input spent by two or more transactions
• Establish consensus on which transactions are valid and the approximate time when they occurred
• Provide transparency by publicly displaying all valid transactions and allowing anyone to keep a copy of and verify the ledger of transactions
• Use the network to derive decentralized trust instead of trusting a single authority
Ledgers in the Traditional Economy

- Kept by banks and government entities (central banks)
- Kept private and not accessible to the public
- Require trust in the government and banks
Structure of a Block

- A block consists of
  - header (metadata)
  - payload (list of transactions)
Header of a Block

- The header of a block includes
  - timestamp
  - two cryptographic hash values
    - hash of the collection of transactions in the block
    - hash of the header of the previous block
  - solution of a cryptographic puzzle (nonce)
    - the inputs of the puzzle include the above hash values
    - the difficulty of the puzzle is adjusted over time
## Connecting Consecutive Blocks

<table>
<thead>
<tr>
<th>time stamp</th>
<th>nonce</th>
<th>trans hash</th>
<th>prev hash</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

transactions

<table>
<thead>
<tr>
<th>time stamp</th>
<th>nonce</th>
<th>trans hash</th>
<th>prev hash</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

transactions
Payload of a Block

- The transactions included in the payload of a block must be
  - New: not already included in previous blocks
  - Valid: pass the verification

- A valid simple transaction has the following properties
  - The public key maps to the input address
  - The signature is verified using the public key
  - The amount transferred to the output address matches the amount in the input address
  - There is no other transaction spending the input bitcoins of this transaction
Adding a Block to the Blockchain

- At any time, there is a collection of new transactions that have not been inserted yet into the blockchain
- Anyone can take a set of new transactions and try to form with them a new block
- This involves the following tasks
  - Collecting new transactions
  - Validating each of them and discarding invalid ones
  - Solving the puzzle associated with the new transactions and the previous block
  - Creating the block header (contains the solution of the puzzle) and payload (contains the new transactions)
Solving the Puzzle

• The puzzle employs a standard cryptographic hash function $h$
• The inputs to the puzzle are
  • $p$: hash of the header of the previous block
  • $c$: hash of the transactions of the current block
  • $t$: target value
• A solution to the puzzle is a nonce (value) $x$ such that
  • $h(x \cdot p \cdot c) < t$
    where $\cdot$ denotes concatenation
• Variant of the Hashcash proof of work (Adam Back, 2002)
Steps in Solving the Puzzle

• Since $h$ is a cryptographic hash function, there is no better strategy to find solution $x$ than brute-force repeated guessing and verifying.

• This involves
  a. generating a guessed solution $x$
  b. concatenating $x$, $p$, and $c$, which yields $y = x \cdot p \cdot c$
  c. evaluating $h$ on $y$ which yields $z = h(y) = h(x \cdot p \cdot c)$
  d. comparing $z$ with the target $t$, i.e., testing if $z < t$

• The evaluation of $h$ in step c is by far the computationally most expensive step.

• The smaller is $t$ the higher is the difficulty of solving the puzzle.
Computational Effort of the Puzzle

• For simplicity, assume the target is a power of 2
  • \( t = 2^{k-h} \), where \( k \) is the number of output bits of the hash function

• Solving the puzzle means finding \( x \) such that hash value \( h(x \cdot p \cdot c) \) has \( h \) leading zeros

• Probability \( G \) of guessing a correct solution
  • \( G = 1/2^h \)
  • Bernoulli trials and expected value of a geometric distribution

• The expected number of guesses until a solution is found is \( 2^h \)

• Solving the puzzles takes about \( 2^h \) hash computations to verify each guessed solution
Bitcoin aims at generating one block every 10 minutes on average.

Every two weeks, the actual block generation rate is reviewed and the difficulty of solving the puzzle may be adjusted:

- if lower than 10 minutes, the target is lowered to increase the difficulty.
- if greater than 10 minutes, the target is increased to lower the difficulty.
- else no changes are made.
Rewards for Adding a Block

- The process of creating a block is computationally expensive
- Most of the work is spent solving the puzzle
- To motivate the addition of a block, bitcoin provides two types of rewards
  - Transaction fee: an optional reward for each transaction included in the block
  - Coinbase: a fixed reward for each block
- These rewards are transferred to an output address specified in the block by the creator of the block
Transaction Fee

- The issuer of a transaction may optionally transfer to the output addresses an amount less than what is available as input.
- The difference, called transaction fee, is automatically transferred to the party creating the block that includes the transaction.
- As of February 2015, the typical transaction fee is 0.0001 BTC, or about a couple of USD pennies.
Coinbase

- So far we have discussed only the transfer of bitcoins
- The coinbase reward enables the creation of new bitcoins and is the only mechanisms that allows to create bitcoin
- The coinbase reward creates new bitcoins that are given to the party adding the block
- The amount of the coinbase reward changes over time
  - Stars at 50 BTC
  - Is halved every 210,000 blocks (about four years)
  - As of February 2015, the coinbase reward is 25 BTC, or about a few thousand USD
Total Bitcoins Created Over Time (millions)

Adaptation of figure by Insti
Wikimedia Commons

Copyright 2015 Giuseppe Di Battista and Roberto Tamassia
Total Bitcoins Created Over Time

- The total number of bitcoin ever created at a given time $t$, denoted $BTC(t)$, can be computed by a formula which is essentially the sum of a geometric series.
- Let $L(t)$ be the length of the blockchain at time $t$.
- Let $Q(t)$ and $R(t)$ be the quotient and remainder of $L(t)$ divided by $210,000$.
- We have
  $$BTC(t) = R(t) \cdot \frac{50}{2^{Q(t)}} + \sum_{i=0}^{Q(t)-1} \left(50 \cdot 210,000 \div 2^i \right)$$
- From the formula, the total number of bitcoin tends to
  $$50 \cdot 210,000 \cdot \sum_{i=0}^{\infty} \frac{1}{2^i} = 50 \cdot 210,000 \cdot 2 = 21,000,000$$
Mining

- The activity associated with adding blocks to the blockchain is called mining
- Mining requires considerable computational resources
- Specialized hardware for mining has been developed
  - Fast hash computations
  - In the early days of bitcoin, mining could be done on a single computer
  - Today mining is done in datacenters
- Miners form groups called mining pools
- Pool members share
  - Computing costs
  - Rewards from mining
Questions

• What prevents the forking of the block chain?
  • Two miners could select different transactions and generate from them different blocks
  • As forks occur, the blockchain would degenerate into a blocktree

• How can consensus on the blockchain be achieved?
  • There is no designated master copy of the blockchain
  • Different parties may see different transactions and different blocks, resulting in different views of the blockchain and different newly created bitcoins
Ensuring Consistency

• Should forking occur and blocks form a tree, each chain (path) from the start node to a leaf block is a candidate to become the consensus blockchain
• Bitcoin establishes that the consensus blockchain is the longest chain (more precisely, the one with largest cumulative puzzle solution difficulty)
• In principle, at any given time, there could be multiple longest chains
• However, the probability of this occurring is very low and most of the time, the longest chain is unique
Instability of Forking

• Since adding a block is a probabilistic puzzle, the probability $p$ of forking with simultaneous block announcements is very low though nonzero

• A repeated occurrence of this event $k$ times has exponentially decreasing probability $p^k$

• If we have instead a fork with blocks announced at different times, it is advantageous for a miner to grow the one announced first, so the other chain gets dropped
Consensus: all the nodes have the same blockchain
Fork: two nodes extend the blockchain at about the same time
The two new blocks propagate through the network
Bitcoin Network and Blockchain Forks

The two new blocks propagate through the network
Bitcoin Network and Blockchain Forks

A node succeeds in extending one of the blockchains
The new block propagates through the network
Bitcoin Network and Blockchain Forks

As the new block propagates, the longer blockchain pushes out the shorter one.
Bitcoin Network and Blockchain Forks

Back to consensus
Confirming Transactions

• Forking may cause a transaction to be invalidated
  • its block may be later discarded
• The likelihood of forking below a block decreases with the distance of the block from the end of the blockchain
• Suppose a transaction is in a block at depth $i$ and let $j \geq i$ be the dept of the current block
• The difference $j - i$, called confirmations, is a measure of confidence that the transaction will remain valid in the future
• It is common to wait for $6$ confirmations (about one hour) for high value transactions
Merkle Tree

- The transactions of a block are hashed using a Merkle tree
- Balanced tree
- Leaves: hashes of individual transactions
- Internal nodes: hierarchical hashing
  - \( a = h(x_1, x_2) \)
  - \( b = h(x_3, x_4) \)
  - \( c = h(a, b) \)
  - ...
- Root hash value stored in the block header
Hash Tree Transaction Verification

• Proof that a transaction is in a block
  • root hash
  • path from the leaf to the root (hash values and L/R indicators)
  • logarithmic size and verification time
• The proof of $x_4$ consists of
  • Root hash $g$
  • Path $[(x_3, L), (a, L), (d, R)]$
• Verification
  • $g = h(h(a, h(x_3, x_4)), d)$
Beyond Bitcoin

- Building services over Bitcoin
- Altcoins
Building Services over Bitcoin

• The blockchain is ideally suited to be used as a timestamping service (e.g., proofofexistence.com)
• Hash a document and create bitcoin address from hash value
• Create a small-value bitcoin transaction payable to the address
  • The transaction will be included in a block i with timestamp t and stored in the blockchain
  • Proof that the document existed at time t
  • Unspendable transaction output
• More properly accomplished with a special transaction
  • OP_RETURN script marks output unspendable
  • Include only transaction fee
Altcoins

- Altcoins are digital currencies based on principles similar to those of Bitcoin
- Often derived from the bitcoin source code (software forks)
- Offer variations of bitcoin parameters and mechanisms
  - Coin creation
  - Proof of work
  - Consensus
- May provide additional or enhanced properties, such as a stronger assurance of anonymity
Litecoin

- Launched in 2011
- Shorter block generation time than bitcoin (faster confirmations)
  - 2 minutes 30 seconds
- Bounded number of coins like bitcoin
- Different proof of work method (Scrypt)
  - memory-hard function that is not parallelizable
    - generate and store a chain of pseudorandom values
    - access them in pseudorandom order
    - return the last value accessed
- About $60M market cap in March 2015
Zerocoin/Zerocash

- Academic project started in 2013
- Teams at Hopkins, MIT, Technion and Tel Aviv
- Privacy-preserving extension of Bitcoin
  - Transactions do not reveal amount or input/output addresses
  - Based on cryptographic commitments and zero-knowledge proofs
- Zerocoin
  - Fully anonymous currency convertible to/from bitcoin
  - Zerocoin transaction cannot be linked