Mining Miscellanea

Saravanan Vijayakumaran
sarva@ee.iitb.ac.in

Department of Electrical Engineering
Indian Institute of Technology Bombay

February 6, 2024

Choosing Between Chain Forks

Difficulty Adjustment

Block Header =

nVersion	4 bytes
hashPrevBlock	32 bytes
hashMerkleRoot	32 bytes
nTime	4 bytes
nBits	4 bytes
nNonce	4 bytes

- Let $b_1 b_2 b_3 b_4$ be the 4 bytes in nBits. The 256-bit target threshold is given by

$$T = b_2 b_3 b_4 \times 256^{b_1 - 3}.$$

- Miner who can find nNonce such that

$$\text{SHA256}\,(\text{SHA256}\,(\text{ nVersion } \| \cdots \| \text{ nNonce })) \leq T$$

  can add a new block
- Every 2016 blocks, the mining target T is recalculated
- Let t_{sum} = Number of seconds taken to mine last 2016 blocks

$$T_{\text{new}} = \frac{t_{\text{sum}}}{2016 \times 10 \times 60} \times T$$

Choose the Most Difficult-to-Produce Chain

- Given a mining target T, the probability of success in a single trial is approximately

$$\frac{T}{2^{256} - 1}$$

- Expected number of hashes to find valid block is $\frac{2^{256}-1}{T}$
- Sum of the expected number of hashes in all blocks in a chain is called its **chainwork**
- Given two valid forks, the Bitcoin nodes choose the chain which has more chainwork
- Remarks
    - Within a difficulty adjustment period, all chains of same length have the same chainwork
    - Forks which span the difficulty transition will have different chainwork

Finding and Distributing Mining Nonces

Bitcoin Mining

Block Header =

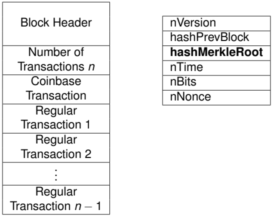
nVersion	4 bytes
hashPrevBlock	32 bytes
hashMerkleRoot	32 bytes
nTime	4 bytes
nBits	4 bytes
nNonce	4 bytes

- A $4000 mining rig can perform 200 TH/s
- A 4-byte nNonce field means $2^{32} \approx 4 \times 10^9$ possibilities
- **What should a miner do if all the 2^{32} nNonce values fail threshold test?**
  - Changing hashPrevBlock and nBits fields invalidates block
  - Change bits in the nVersion field?
  - Change timestamp to change nTime field?
  - Change transactions to change hashMerkleRoot field?

Modifying nVersion and nTime

- nVersion
  - Three bits of the 32-bit nVersion are set to 001
  - Remaining 29 bits are used by miners to signal support for soft forks
  - Changing the signaling bits can interfere with protocol upgrades
  - Some miners still do it (see block 541,604)
- nTime
  - Timestamps can be changed only by increments of a second
  - In block at height N, the nTime value needs to be greater than median of nTime values of blocks $N-1, N-2, \ldots, N-11$
  - A node rejects a block if the nTime field specifies a time which exceeds its network-adjusted time by more than 2 hours
  - Miners cannot risk invalidating their mined blocks by modifying nTime indiscriminately

Transaction Merkle Root



Block Header

Number of Transactions n

Coinbase Transaction

Regular Transaction 1

Regular Transaction 2

$\vdots$

Regular Transaction $n-1$

nVersion

hashPrevBlock

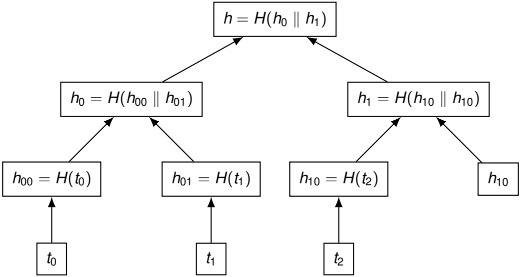
hashMerkleRoot

nTime

nBits

nNonce

- hashMerkleRoot contains root hash of transaction Merkle tree
- Modifying any transaction or the transaction order will modify the root hash



$h = H(h_0 \parallel h_1)$

$h_0 = H(h_{00} \parallel h_{01})$

$h_1 = H(h_{10} \parallel h_{10})$

$h_{00} = H(t_0)$

$h_{01} = H(t_1)$

$h_{10} = H(t_2)$

h_{10}

t_0

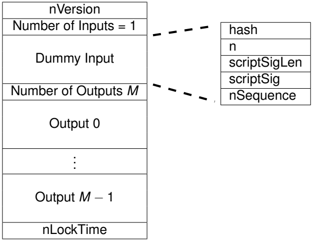
t_1

t_2

The Extra Nonce Solution

- Although coinbase transaction do not unlock previous outputs, they contain a dummy input

Coinbase Transaction Format



nVersion

Number of Inputs = 1

Dummy Input

Number of Outputs M

Output 0

$\vdots$

Output $M-1$

nLockTime

hash

n

scriptSigLen

scriptSig

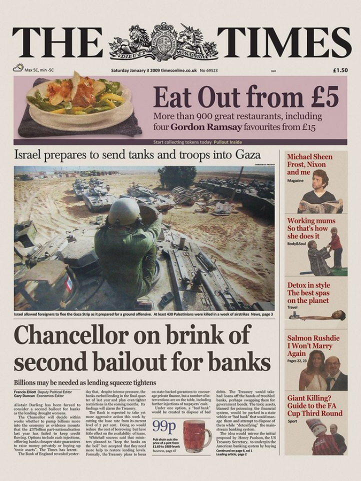
nSequence

- Dummy input fields
  - hash is set to all zeros (0x000. . . 000)
  - n is set to 0xFFFFFFFF
  - scriptSig field can be at most 100 bytes long; also called coinbase field
  - Since March 2013, the first 4 bytes of scriptSig encode the block height
  - The remaining scriptSig space is used as an **extra nonce** by miners

Genesis Block Coinbase Field

- Satoshi put the following text in the genesis block coinbase field

The Times 03/Jan/2009 Chancellor on brink of second bailout for banks



THE TIMES

Max 5C, min -5C

Saturday January 3 2009 timesonline.co.uk No 69523

£1.50

Eat Out from £5

More than 900 great restaurants, including four **Gordon Ramsay** favourites from £15

Start collecting tokens today Pullout Inside

Israel prepares to send tanks and troops into Gaza

Israel allowed foreigners to flee the Gaza Strip as it prepared for a ground offensive. At least 430 Palestinians were killed in a week of airstrikes News, page 3

Chancellor on brink of second bailout for banks

Billions may be needed as lending squeeze tightens

Francis Elliott Deputy Political Editor
Gary Duncan Economics Editor

Alistair Darling has been forced to consider a second bailout for banks as the lending drought worsens.

The Chancellor will decide within weeks whether to pump billions more into the economy as evidence mounts that the £37billion part-nationalisation last year has failed to keep credit flowing. Options include cash injections, offiering banks cheaper state guarantees to raise money privately or buying up "toxic assets", The Times has learnt.

The Bank of England revealed yesterday that, despite intense pressure, the banks curbed lending in the final quarter of last year and plan even tighter restrictions in the coming months. Its findings will alarm the Treasury.

The Bank is expected to take yet more aggressive action this week by cutting the base rate from its current level of 2 per cent. Doing so would reduce the cost of borrowing but have little effect on the availability of loans.

Whitehall sources said that ministers planned to "keep the banks on the boil" but accepted that they need more help to restore lending levels. Formally, the Treasury plans to focus on state-backed gurantees to encourage private finance, but a number of interventions are on the table, including further injections of taxpayers' cash.

Under one option, a "bad bank" would be created to dispose of bad debts. The Treasury would take bad loans off the hands of troubled banks, perhaps swapping them for government bonds. The toxic assets, blamed for poisoning the financial system, would be parked in a state vehicle or "bad bank" that would manage them and attempt to dispose of them while "detoxifying" the mainstream banking system.

The idea would mirror the initial proposal by Henry Paulson, the US Treasury Secretary, to underpin the American banking system by buying

Continued on page 6, col 1
Leading article, page 2

99p

Pub chain cuts the price of a pint from £1.69 to 1989 levels

Business, page 47

Michael Sheen Frost, Nixon and me

Magazine

Working mums So that's how she does it

Body&Soul

Detox in style The best spas on the planet

Travel

Salmon Rushdie I Won't Marry Again

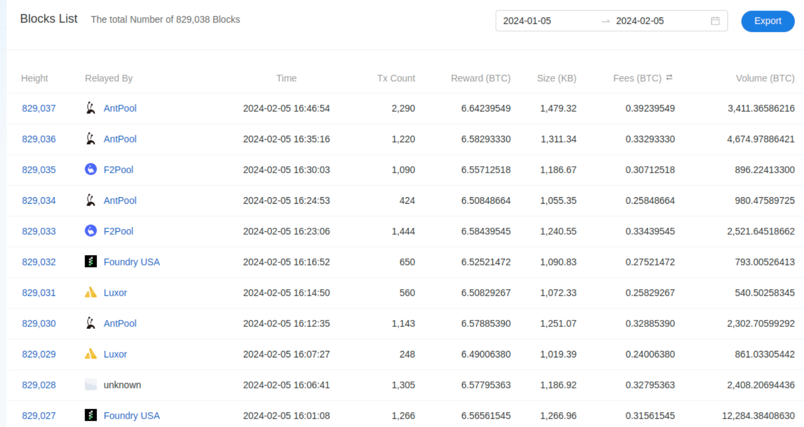
Pages 22, 23

Giant Killing? Guide to the FA Cup Third Round

Sport

Coinbase Markers

- Miners identify themselves in the coinbase field



Height	Relayed By	Time	Tx Count	Reward (BTC)	Size (KB)	Fees (BTC) ⇄	Volume (BTC)
829,037	AntPool	2024-02-05 16:46:54	2,290	6.64239549	1,479.32	0.39239549	3,411.36586216
829,036	AntPool	2024-02-05 16:35:16	1,220	6.58293330	1,311.34	0.33293330	4,674.97886421
829,035	F2Pool	2024-02-05 16:30:03	1,090	6.55712518	1,186.67	0.30712518	896.22413300
829,034	AntPool	2024-02-05 16:24:53	424	6.50848664	1,055.35	0.25848664	980.47589725
829,033	F2Pool	2024-02-05 16:23:06	1,444	6.58439545	1,240.55	0.33439545	2,521.64518662
829,032	Foundry USA	2024-02-05 16:16:52	650	6.52521472	1,090.83	0.27521472	793.00526413
829,031	Luxor	2024-02-05 16:14:50	560	6.50829267	1,072.33	0.25829267	540.50258345
829,030	AntPool	2024-02-05 16:12:35	1,143	6.57885390	1,251.07	0.32885390	2,302.70599292
829,029	Luxor	2024-02-05 16:07:27	248	6.49006380	1,019.39	0.24006380	861.03305442
829,028	unknown	2024-02-05 16:06:41	1,305	6.57795363	1,186.92	0.32795363	2,408.20694436
829,027	Foundry USA	2024-02-05 16:01:08	1,266	6.56561545	1,266.96	0.31561545	12,284.38408630

Source: `https://explorer.btc.com/btc/blocks`

Block Distribution

- The percentage of blocks mined by each miner can be calculated from coinbase markers

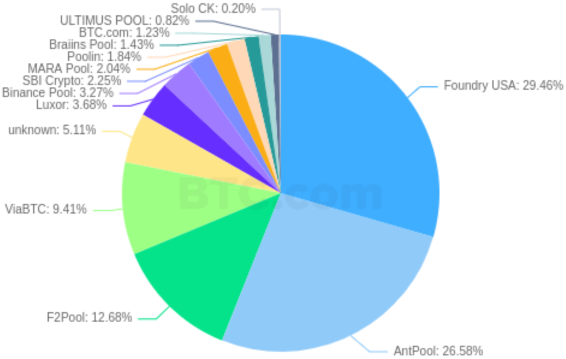


Image credit: `https://explorer.btc.com/btc/insights-pools`

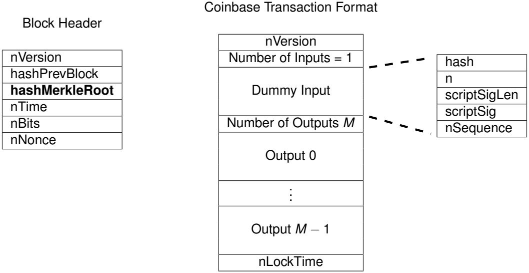
Mining Pools

- The network hashrate is 500 Exahashes/s = 500×10^{18} hashes/s
- A $4000 mining rig can perform 200 TH/s
- The probability of an individual rig owner winning a block is low
- Rig owners join mining pools
- Mining pool operation
    - Pool owner “distributes” the mining search space among the pool miners (participants)
    - When a pool miner finds a hash starting with 32 zeros, it submits the block header to the pool as proof of its efforts. This is called a **share**.
    - If one of the pool miners finds a valid block, the block reward is distributed to all pool miners proportional to the number of submitted shares
    - Pool takes a portion of the block reward as coordination fee

Distributing Search Space

- Pool owner can distribute search space by having a different extra nonce for each pool miner
- Rolling of extra nonce by pool owner for every pool miner does not scale
    - Pool owner recomputes hashMerkleRoot for every extra nonce change
    - Pool miners only change nNonce and nTime (assuming nVersion is not changed)
- Instead, extra nonce is split into two parts
    - ExtraNonce1 is used to distribute search space
    - ExtraNonce2 is changed by the individual pool miners

Transaction Merkle Root



Block Header

nVersion
hashPrevBlock
hashMerkleRoot
nTime
nBits
nNonce

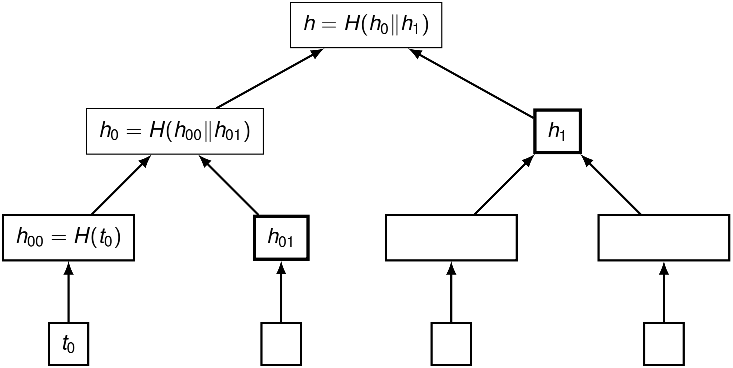
Coinbase Transaction Format

nVersion
Number of Inputs = 1
Dummy Input
Number of Outputs M
Output 0
⋮
Output $M - 1$
nLockTime

hash
n
scriptSigLen
scriptSig
nSequence

- Pool owner sends each pool miner the following
    - nVersion, hashPrevBlock, nTime, nBits fields of block header
    - Coinbase1 = Part of the coinbase transaction before extra nonce
    - ExtraNonce1 = Miner-specific extra nonce
    - ExtraNonce2_size = The number of bytes in ExtraNonce2 the miner can change
    - Coinbase2 = Part of the coinbase transaction after extra nonce
    - Merkle_branch = List of hashes used to calculate hashMerkleRoot

Merkle Branch



- Every time ExtraNonce2 is changed, the hashMerkleRoot has to be recalculated
- Instead of sending all the transactions, only necessary hashes are sent

AsicBoost

SHA-256

- SHA = Secure Hash Algorithm, 256-bit output length
- Accepts bit strings of length upto $2^{64} - 1$
- Output calculation has two stages
  - Preprocessing
  - Hash Computation
- Preprocessing
  1. A 256-bit state variable $H^{(0)}$ is set to

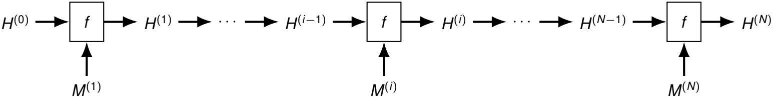
$$\begin{aligned} H_0^{(0)} &= \mathtt{0x6A09E667}, & H_1^{(0)} &= \mathtt{0xBB67AE85}, \\ H_2^{(0)} &= \mathtt{0x3C6EF372}, & H_3^{(0)} &= \mathtt{0xA54FF53A}, \\ H_4^{(0)} &= \mathtt{0x510E527F}, & H_5^{(0)} &= \mathtt{0x9B05688C}, \\ H_6^{(0)} &= \mathtt{0x1F83D9AB}, & H_7^{(0)} &= \mathtt{0x5BE0CD19}. \end{aligned}$$

  2. The input M is padded to a length which is a multiple of 512

SHA-256 Hash Computation

1. Padded input is split into N 512-bit blocks $M^{(1)}, M^{(2)}, \ldots, M^{(N)}$
2. Given $H^{(i-1)}$, the next $H^{(i)}$ is calculated using a function f

$$H^{(i)} = f(M^{(i)}, H^{(i-1)}), \quad 1 \leq i \leq N.$$



3. f is called a *compression function*
4. $H^{(N)}$ is the output of SHA-256 for input M

SHA-256 Compression Function Building Blocks

- U, V, W are 32-bit words
- $U \wedge V$, $U \vee V$, $U \oplus V$ denote bitwise AND, OR, XOR
- $U + V$ denotes integer sum modulo 2^{32}
- $\neg U$ denotes bitwise complement
- For $1 \leq n \leq 32$, the shift right and rotate right operations

$$\mathrm{SHR}^n(U) = \underbrace{000 \cdots 000}_{n \text{ zeros}} u_0 u_1 \cdots u_{30-n} u_{31-n},$$

$$\mathrm{ROTR}^n(U) = u_{31-n+1} u_{31-n+2} \cdots u_{30} u_{31} u_0 u_1 \cdots u_{30-n} u_{31-n},$$

- Bitwise choice and majority functions

$$\mathrm{Ch}(U, V, W) = (U \wedge V) \oplus (\neg U \wedge W),$$
$$\mathrm{Maj}(U, V, W) = (U \wedge V) \oplus (U \wedge W) \oplus (V \wedge W),$$

- Let

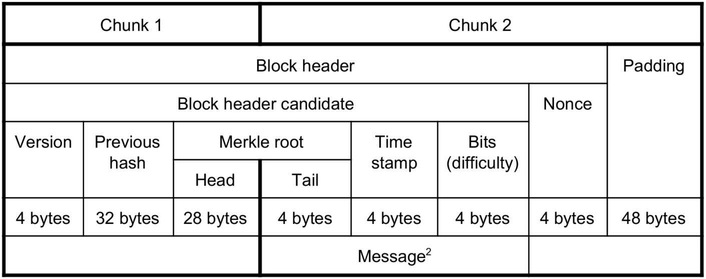
$$\Sigma_0(U) = \mathrm{ROTR}^2(U) \oplus \mathrm{ROTR}^{13}(U) \oplus \mathrm{ROTR}^{22}(U)$$
$$\Sigma_1(U) = \mathrm{ROTR}^6(U) \oplus \mathrm{ROTR}^{11}(U) \oplus \mathrm{ROTR}^{25}(U)$$
$$\sigma_0(U) = \mathrm{ROTR}^7(U) \oplus \mathrm{ROTR}^{18}(U) \oplus \mathrm{SHR}^3(U)$$
$$\sigma_1(U) = \mathrm{ROTR}^{17}(U) \oplus \mathrm{ROTR}^{19}(U) \oplus \mathrm{SHR}^{10}(U)$$

SHA-256 Compression Function Calculation

- Maintains internal state of 64 32-bit words $\{W_j \mid j = 0, 1, \ldots, 63\}$
- Also uses 64 constant 32-bit words $K_0, K_1, \ldots, K_{63}$ derived from the first 64 prime numbers $2, 3, 5, \ldots, 307, 311$
- $f(M^{(i)}, H^{(i-1)})$ proceeds as follows
  1. Internal state initialization
  $$W_j = \begin{cases} M_j^{(i)} & 0 \le j \le 15, \\ \sigma_1(W_{j-2}) + W_{j-7} + \sigma_0(W_{j-15}) + W_{j-16} & 16 \le j \le 63. \end{cases}$$
  2. Initialize eight 32-bit words
  $$(A, B, C, D, E, F, G, H) = \left(H_0^{(i-1)}, H_1^{(i-1)}, \ldots, H_6^{(i-1)}, H_7^{(i-1)}\right).$$
  3. For $j = 0, 1, \ldots, 63$, iteratively update $A, B, \ldots, H$
  $$\begin{aligned} T_1 &= H + \Sigma_1(E) + \mathrm{Ch}(E, F, G) + K_j + W_j \\ T_2 &= \Sigma_0(A) + \mathrm{Maj}(A, B, C) \\ (A, B, C, D, E, F, G, H) &= (T_1 + T_2, A, B, C, D + T_1, E, F, G) \end{aligned}$$
  4. Calculate $H^{(i)}$ from $H^{(i-1)}$
  $$(H_0^{(i)}, H_1^{(i)}, \ldots, H_7^{(i)}) = \left(A + H_0^{(i-1)}, B + H_1^{(i-1)}, \ldots, H + H_7^{(i-1)}\right).$$

AsicBoost

- A method to speedup Bitcoin mining by a factor of 20%
- Proposed by Timo Hanke and Sergio Demian Lerner
- Exploits the fact that SHA256 operates on 64 byte chunks
- The Bitcoin block header is 80 bytes long



<table>
<tr><td colspan="3">Chunk 1</td><td colspan="5">Chunk 2</td></tr>
<tr><td colspan="7">Block header</td><td rowspan="4">Padding</td></tr>
<tr><td colspan="6">Block header candidate</td><td rowspan="3">Nonce</td></tr>
<tr><td rowspan="2">Version</td><td rowspan="2">Previous hash</td><td colspan="2">Merkle root</td><td rowspan="2">Time stamp</td><td rowspan="2">Bits (difficulty)</td></tr>
<tr><td>Head</td><td>Tail</td></tr>
<tr><td>4 bytes</td><td>32 bytes</td><td>28 bytes</td><td>4 bytes</td><td>4 bytes</td><td>4 bytes</td><td>4 bytes</td><td>48 bytes</td></tr>
<tr><td colspan="3"></td><td colspan="3">Message^2</td><td colspan="2"></td></tr>
</table>

Image source: `https://arxiv.org/abs/1604.00575`

- If two transaction Merkle roots collide in the last 4 bytes, some of the SHA-256 work in the second chunk can be reused
- Recall that the internal state initialization (W_j calculation) does not depend on the previous hash $H^{(i-1)}$

AsicBoost Loop

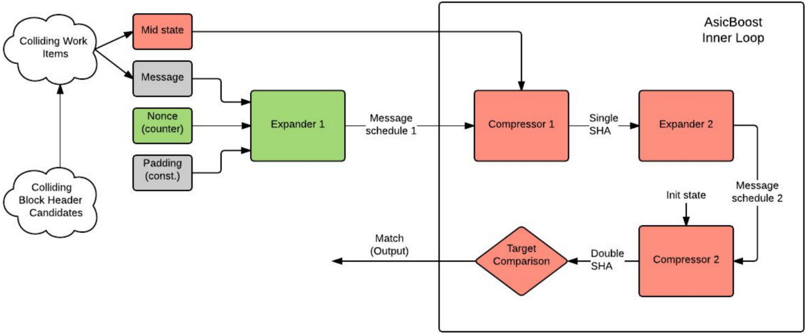


Image source: `https://arxiv.org/abs/1604.00575`

- In the above figure, the grey and green blocks represent computation that can be reused
- If two transaction Merkle roots coincide in the last 4 bytes, then the output of Expander 1 can be reused

References

- What is chainwork? https://bitcoin.stackexchange.com/questions/26869/what-is-chainwork/26894
- Sections 4.2, 4.3, 5.3 of *An Introduction to Bitcoin*, S. Vijayakumaran, www.ee.iitb.ac.in/~sarva/bitcoin.html
- BIP 34: Block v2, Height in Coinbase https://github.com/bitcoin/bips/blob/master/bip-0034.mediawiki
- Bitcoin Genesis Block https://en.bitcoin.it/wiki/Genesis_block
- Bitcoin Blocks with Coinbase Markers https://btc.com/block
- Bitcoin Block Distribution https://btc.com/stats/pool
- Bitmain Mining Rigs https://shop.bitmain.com/
- Slushpool Documentation https://slushpool.com/help/hashrate-proof/
- Hardening Stratum, the Bitcoin Pool Mining Protocol https://arxiv.org/abs/1703.06545
- AsicBoost https://arxiv.org/abs/1604.00575