

The Mathematics Behind Bitcoin Double Spend Race

CYRIL GRUNSPAN

de Vinci Finance Group
ESILV

AND RICARDO PEREZ-MARCO

CNRS
University Paris 7

September 09th 2020

1 Historical roots

ca 1800 BCE. Sîn-kašid, tablets, **centralized government** and harmonisation of **tax**. Silver shekel, units of weight or currency.

[ca 680 BCE — ca 547 BCE]. Mermnad dynasty: Gyges, Ardys, Sadyattes II, Alyattes, Croesus, Lydia, Pactolus, **Histories**, Herodotus of Halicarnasse, (ca. 450). Gold coins, Artemision, Ephesus. **Manipulation** of the state.

1494. Venice, **Tractatus XI Particularis de Computibus et Scripturis**, Luca Pacioli, invention of **Double-Entry Bookkeeping System**.

1923. **Hyperinflation** in the Weimar Republic. Solved by Hjalmar Schacht. Must read : **Le Banquier du Diable**, J.-F. Bouchard, Max Milo (2015)

Europe. Individualism.

Italy: 10 cents Botticelli, 1€ Da Vinci, 2€ Dante, Spain : Cervantès (10, 20, 50 cents)

Importance of middlemen in business, clearing houses...

Daniel Pinto :

To sell a loan is a very cumbersome, time-consuming process; settlement can take weeks.

2 Political roots

Cypherpunks electronic mailing list, Tim May
[The Crypto Anarchist Manifesto](#), T. May (1992)

Openbazar, decentralized e-commerce open source project,
Sam Patterson

PGP Zimmerman

Satoshi Nakamoto satoshi at vistomail.com
Thu Nov 6 15:15:40 EST 2008

*>You will not find a solution to
political problems in cryptography.*

Yes, but we can win a major battle
in the arms race and gain **a new
territory of freedom for several
years.**

Governments are good at cutting off
the heads of a centrally controlled
networks like Napster, but pure P2P
networks like Gnutella and Tor seem to
be holding their own.

Satoshi

The Cryptography Mailing List

[www.metzdowd.com/pipermail/cryptography/2008-
November/014823.html](http://www.metzdowd.com/pipermail/cryptography/2008-November/014823.html)

3 Advances in computer science

TCP/IP

Transmission Control Protocol / Internet Protocol. V.

G. Cerf, B. Kahn, 1972

Open, shared public network without any central authority

Secure and scalable

Arpanet

Internet

E-mail

World Wide Web, R. Cailliau, T. Berners-Lee, 1987

Chat online, Instant messaging

P2P computer networks

≠ client-server model

Music-sharing application Napster (1999 – 2001)

BitTorrent, Peer-to-peer file sharing

Distributed systems

Byzantine fault tolerance

NASA late 70s.

[The Byzantine Generals Problem](#), L. Lamport, R. Shostak, M. Pease, ACM Transactions on Programming Languages and Systems (1982)

[Another Advantage of Free Choice: completely asynchronous agreement protocols](#), M. Ben-Or (1983)

Randomized Consensus

Paxos, Lamport (1989)

King Algorithm

ZZZZYVA

4 Bitcoin References

- [1]. W.Dai,“b-money,”
<http://www.weidai.com/bmoney.txt>, 1998.
- [2]. H. Massias, X.S. Avila, and J.-J. Quisquater, “Design of a secure timestamping service with minimal trust requirements,” In 20th Symposium on Information Theory in the Benelux, May 1999
- [3]. S. Haber, W.S. Stornetta, “How to time-stamp a digital document,” In Journal of Cryptology, vol 3, no 2, pages 99-111, 1991.
- [4]. D. Bayer, S. Haber, W.S. Stornetta, “Improving the efficiency and reliability of digital time-stamping,” In Sequences II: Methods in Communication, Security and Computer Science, pages 329-334, 1993.
- [5]. S. Haber, W.S. Stornetta, “Secure names for bit-strings,” In Proceedings of the 4th ACM Conference on Computer and Communications Security, pages 28-35, April 1997.
- [6]. A. Back, “Hashcash - a denial of service counter-measure,”
<http://www.hashcash.org/papers/hashcash.pdf>, 2002.
- [7]. R.C. Merkle, “Protocols for public key cryptosystems,” In Proc. 1980 Symposium on Security and Privacy, IEEE Computer Society, pages 122-133, April 1980.
- [8]. W. Feller, “An introduction to probability theory and its applications,” 1957.

5 Advances in Cryptography

5.1 Public key cryptography

UK, seventies, J.H. Ellis, C. Cocks, M.J. Williamson
Research declassified by the British government in 1997.

Diffie–Hellman key exchange

New Directions in Cryptography, W. Diffie, M. Hellman, IEE Transactions on Information theory (1976).

1977, R. Rivest, A. Shamir, L. Adleman

Alice and Bob

Secret key, public key

RSA

ECDSA

Elliptic curve on Galois field \mathbb{F}_p secp256k1,

$$y^2 = x^3 + 7$$

with

$$\begin{aligned} p &= 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1 \\ &= 11579208923731619542357098500868790785326998466\backslash \\ &\quad 5640564039457584007908834671663 \end{aligned}$$

Only used for Bitcoin? Gaining in popularity.

Elliptic curve useful for generating a finite group

Discrete logarithmic problem hard to solve

Base point G

Secret integer n

Public key $= n \cdot G$

5.2 Hash functions

Rabin, Yuval, Merkle, late 70.

“Swiss army knife” of cryptography

- input of any size
- output of fixed-size
- easy to calculate (in $O(n)$ if input is n -bit string)
 - i. collision resistance
 - ii. preimage resistance
 - iii. second preimage resistance

One way function

Random Oracles are Practical: A Paradigm for Designing Efficient Protocols, M. Bellare, P. Rogaway, ACM Conference on Computer and Communications Security (1993).

Based on block ciphers

Compression function

Initialization Vector (IV)

Merkle–Damgård construction

Birthday paradox

Integrity of transferred data

Message digest

Commitments

Puzzle

Digital signature

SHA-1, MD5 broken

SHA-2

5.3 Proof of Work

Use of hash function to create a puzzle

Time consuming

Cost function. A string, D integer, x integer

$$\begin{aligned}\mathcal{F}: \quad \mathcal{C} \times [0, D_{\max}] \times [0, N] &\longrightarrow \{\text{True}, \text{False}\} \\ (A, D, x) &\longmapsto \mathcal{F}(A, D, x)\end{aligned}$$

Example: $\mathcal{F}(A, D, x) = \text{True}$ if $\text{Hash}(A|D|x)$ starts with D zeros and false else.

Problem. Given A, D , find \mathbf{x} such that

$$\mathcal{F}(A, D, \mathbf{x}) = \text{True} \tag{1}$$

Solution \mathbf{x} (not necessarily unique) called **nonce**

Very hard to solve

Use of computational power

Pricing via Processing or Combatting Junk Mail, C. Dwork and M. Naor, (1993).

Denial-of-service counter measure technique in a number of systems

Anti-spam tool

Hashcash, A Denial of Service Counter-Measure, A. Back, preprint (2002)

Hashcash: a proof-of-work algorithm

Create a stamp to attach to mail

Cost functions proposed are different

Solution of (1) by brute-force.

Calculus of plenty of hash

5.4 Merkle root

Patent in 1979...

[A Digital Signature Based on a Conventional Encryption Function](#), R. C. Merkle (1988).

Merkle tree = Tree of hashes

Oriented Acyclic Rooted tree

Binary Tree

Leaf = Hash (block)

Top Hash = Merkle root

Used to check integrity of a list of blocks

How to prove that an element x belongs to a set S ?

Screen all S ? Solution in $O(n)$.

Solution proportional to the logarithm of the number of nodes of the tree $O(\ln(n))$

Any permutation of leaves gives a new Merkle root...

5.5 Timestamping

Works of S. Haber, W.S. Stornetta, J.-J. Quisquater,...

How should a Patent Office timestamp a digital document?

Let D be a document. How to define $\text{Certificate}(D)$?

$$\text{Certificate}(D) := \text{Hash}(D) ?$$

If D came just before D' how to prove it using certificates?

Idea: Certificate of D reused to define Certificate of D'

$$\text{Certificate}(D') := \text{Hash}(D' | \text{Certificate}(D))$$

Proof that D came before...

Improving the Efficiency and Reliability of Digital Time-Stamping, D. Bayer, S. Haber, W.S. Stornetta (1993)

To establish that a document was created after a given moment in time, it is necessary to report events that could not have been predicted before they happened. To establish that a document was created before a given moment in time, it is necessary to cause an event based on the document, which can be observed by others.

What if many documents came at the same period of time? Solution with a Merkle tree...

6 “Blockchain”

Block of certificates

Each block contains a reference to previous block

Ledger of certificates.

Block = Merkle tree of certificates + header(block)

Header(Block) = Merkle root + Hash(previous block header) + (possibly a date?)

Easy to check if a given certificate belongs to the ledger

Any modification of the ledger is automatically detected

Linked list of hash pointers

Huge difference between blockchain and proof of work

Concepts are different!

Blockchain popularized by bitcoin

Blockchain = Ledger

Hal Finney on the cypherpunk mailing list

7 The creation of Bitcoin

Bitcoin: A Peer-to-Peer Electronic Cash System,
October 31, 2008.

Probably more cryptography in payment cards than
in Bitcoin!

ECDSA

Secret/Public Key with secp256k1

Hash Functions RIPEMD160 & SHA-256.

Bitcoin Address = $\text{SHA-256} \circ \text{RIPEMD160}(\text{PublicKey})$

Each block contains a Merkle tree of transactions

Blockchain, Ledger

Proof of work for mining blocks

Two natural questions

1. How to avoid double spending?
2. How can it work in a decentralized network?

Two clever answers

1. Make each transaction a patent certificate
2. Use of proof of work to get decentralization

Tour de force

More related questions.

1. How to avoid Sybill attacks?
2. How to solve the byzantine problem?

Answers

1. Naturally thanks to proof of work
2. Thanks to public key cryptography & assumption that “The requirement is that the good guys collectively have more CPU power than any single attacker.”

Commitment schemes in Lightning Network
Payment Channels

Many failed attempts

- SET (Visa & Mastercard)
- Cybercash (bug 2000)
- Bitgold (Nick Szabo, 1998)
- Digicash (David Chaum 1990, banqueroute 1998)
- B-money (Wei Dai 1998)
- Paypal (1998)

Blind signatures for untraceable payments, David Chaum (1983)

Satoshi Nakamoto (forum **bitcointalk** 2010) :

Bitcoin is an implementation of Wei Dai's b-money proposal on Cypherpunks in 1998 and Nick Szabo's Bitgold proposal.

Satoshi's white paper

Double-spending is prevented with a peer-to-peer network. No mint or other trusted parties. Participants can be anonymous. New coins are made from Hashcash style proof-of-work. The proof-of-work for new coin generation also powers the network to prevent double-spending.

Everything is public

Ledger of transactions

Page = block

Everybody can maintain the ledger

Writer = miner

Money transfer = smart contract

Satoshi

The only way to prevent double spending is to have a ledger accounting for all transactions, so that the recipient can check that the transaction is legitimate. If we don't want this ledger to be centralized under the control of a third party, then it must be public.

What is a (bit)coin?

We define an electronic coin as a chain of digital signatures. Each owner transfers the coin to the next by digitally signing a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin. A payee can verify the signatures to verify the chain of ownership.

Transaction = 2 scripts = scriptsig + scriptpubkey

What does a miner do?

Verify transactions, gather valid transactions, constitute a block, he tries to win the mining race with other miners. The first one to mine a new block wins 12.5 bitcoins.

How to recognize the official blockchain?

It is $(B_i)_{0 \leq i \leq N}$ such that $\sum_{i=0}^N D_i$ is maximum with $D_i = \text{difficulty associated with block } B_i$.

Difficulty adjusted every 2016 blocks

Official blockchain \approx longest chain

8 Why should we trust Bitcoin?

8.1 First results

Satoshi was wrong !

Underestimation of double spend success probability

Existence of closed form formulas

Mathematical foundation of Bitcoin

Bitcoin and Gamma functions

Notation 1. Let $0 < q < \frac{1}{2}$ (resp. $p = 1 - q$), the relative hash power of the group of attackers (resp. of honest miners).

Theorem 2. After z blocks have been validated by the honest miners, the probability of success of the attackers is

$$P(z) = I_{4pq} \left(z, \frac{1}{2} \right)$$

where $I_x(a, b)$ is the *regularized incomplete beta function*

$$I_x(a, b) := \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \int_0^x t^{a-1} (1-t)^{b-1} dt$$

Corollary 3. Let $s = 4pq < 1$. When $z \rightarrow \infty$, we have

$$P(z) \sim \frac{s^z}{\sqrt{\pi(1-s)} s}$$

8.2 Other results

Given $z \in \mathbb{N}$, block generation time t for mining z block(s) is publicly known.

Definition 4. We denote by $P(z, t)$ the probability of success of a double spend attack when z blocks have been validated within a period of time of t .

What we'll obtain also:

- Closed form formula for $P(z, t)$.
- Satoshi's formula $P_{\text{SN}}(z)$ is actually a $P(z, t)$
- Asymptotics formulas for $P_{\text{SN}}(z)$ and $P(z, t)$
- Explicit rank z_0 such that $P(z) < P_{\text{SN}}(z)$.

In particular,

$$P_{\text{SN}}(z) \sim \frac{e^{-z\left(\frac{q}{p}-1-\ln\frac{q}{p}\right)}}{2}$$

9 Other considerations

Economic cost of a double spend attack (with and without “eclipse attack”):

[An Analysis of Attacks on Blockchain Consensus](#)

G. Bissias, B. Levine, A. Pinar-Pzisik, G. Andresen, preprint, (2016/11/20).

Is it necessary to wait for confirmations?

Have a Snack, Pay With Bitcoins

O. Bamert, C. Decker, L. Elsen, R. Wattenhofer, S. Welten, 13-th IEEE International Conference on Peer-to-Peer Computing, Trento, Italy (2013).

10 Mathematics of mining

10.1 Introduction

Looking for a nonce y (“number used once”)
= Waiting for a bus!

Hashcash proof-of-work (Adam Back).

F = hash function = SHA256²

Looking for y such that $F(x|y) < \text{Target}$

$x = x1|x2|x3|x4|x5$

$x1$ = Version

$x2$ = hash Previous Block

$x3$ = hash Merkle Root

$x4$ = Timestamp

$x5$ = Target

Block Header = $x|y$.

Reference: Bitcoin Wiki

https://en.bitcoin.it/wiki/Block_hashing_algorithm

Example 5. Block Hash 0

000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f

Example 6. Block Hash 447384

0000000000000000000027175e4c9a3216c1331650e45eafdb948ff03ab59ef1778

Notation 7. (*Random variable*) Interblock Time is \mathbf{T} . Time used for mining k -th block \mathbf{T}_k .

Bitcoin Protocol : $\mathbb{E}[\mathbf{T}] = \tau_0 := 600$ (seconds)

Adjustment of target every 2 weeks ($2016 = 2 \times 7 \times 24 \times 6$ blocs)

See <https://bitcoinwisdom.com/bitcoin/difficulty>

$$\text{Target}_{\text{new}} = \text{Target}_{\text{old}} \cdot \frac{t}{2016 \times \tau_0}$$

where t = time spent for mining the last 2016 blocks.

10.2 Mining one block

The time it takes to mine a block is memoryless

$$\mathbb{P}[T > t_1 + t_2 | T > t_2] = \mathbb{P}[T > t_1]$$

Proposition 8. *The random variable \mathbf{T} has the **exponential distribution** with parameter $\alpha = \frac{1}{600}$ i.e.,*

$$f_{\mathbf{T}}(t) = \alpha e^{-\alpha t}$$

Parameter α seen as a **mining speed**, $\mathbb{E}[\mathbf{T}] = \frac{1}{\alpha}$.
Confirmation by studying timestamps sequence

10.3 Mining more blocks

Interblock times $\mathbf{T}_1, \dots, \mathbf{T}_n$ are **independent identically distributed** exponential random variables. The sum

$$\mathbf{S}_n = \mathbf{T}_1 + \dots + \mathbf{T}_n$$

is the time spent to get n blocks

Proposition 9. *The random variable \mathbf{S}_n has a **Gamma distribution** with parameter (n, α) :*

$$f_{\mathbf{S}_n}(t) = \frac{\alpha^n}{(n-1)!} t^{n-1} e^{-\alpha t}$$

Definition 10. Let $\mathbf{N}(t)$ be the number of blocks already mined at t -time. Start is at $t=0$.

Proposition 11. The random process \mathbf{N} is a *Poisson process* with parameter α i.e.,

$$\mathbb{P}[\mathbf{N}(t) = k] = \frac{(\alpha t)^k}{k!} e^{-\alpha t}$$

Notation 12. The letters $\mathbf{T}, \alpha, \mathbf{S}_n, \mathbf{N}$ (resp. $\mathbf{T}', \alpha', \mathbf{S}'_n, \mathbf{N}$) are reserved for honest miners (resp. attacker).

10.4 Interpretation of speed mining

Same notations as above. Mining speed α (honest) and α' (attacker). Probability p (honest) and q (attacker). We note also $\tau_0 = 600$ seconds = 10 minutes.

Proposition 13. We have:

$$p = \mathbb{P}[\mathbf{T} < \mathbf{T}'] \tag{2}$$

$$p = \frac{\alpha}{\alpha + \alpha'} \tag{3}$$

$$q = \frac{\alpha'}{\alpha + \alpha'} \tag{4}$$

$$\alpha + \alpha' = \frac{1}{\tau_0} \tag{5}$$

$$\alpha = \frac{p}{\tau_0} \tag{6}$$

$$\alpha' = \frac{q}{\tau_0} \tag{7}$$

Proof. The random variable $\text{Inf}(\mathbf{T}, \mathbf{T}')$ has the exponential distribution with parameter $\alpha + \alpha'$. \square

Proof. (Another proof). Denote by h (resp. h') the hashrate of the honest miners (resp. attacker) and t_0 (resp. t'_0) the average time it takes for mining a block.

Total hashrate of the network $= h + h'$.

Proof-of-work: search for a **nonce** in Block Header such that

$$\text{Hash}(\text{Block Header}) < \text{Target}$$

Set $m = \frac{2^{256}}{\text{Target}}$ We have

$$p = \frac{h}{h + h'} \quad (8)$$

$$q = \frac{h'}{h + h'} \quad (9)$$

$$(h + h') \tau_0 = m \quad (10)$$

$$h t_0 = m \quad (11)$$

$$h' t'_0 = m \quad (12)$$

□

So, α, h, p are proportionnal.

11 Classical Double Spend Attack

No eclips attack

11.1 What is a double spend?

A single output may not be used as an input to multiple transactions.

- $T = 0$. A merchant **M** receives a transaction **tx** from **A** (= attacker). Transaction **tx** is issued from an UTXO **tx0**
- Honest Miners start mining **openly, transparently**
- Attacker **A** starts mining **secretly**
- One block of honest miners include **tx**
- No block of attacker include **tx**
- On the contrary, one blocks of the attacker includes another transaction **tx'** conflicting with **tx** from same UTXO **tx0**
- As soon as the z -th block has been mined, **M** sends his good to **A**
- **A** keeps on mining secretly
- As soon as **A** has mined a blockchain with a lenght greater than the official one, **A** releases his blockchain to the network
- Transaction **tx** has disappeared from the official blockchain.

Free Lunch!

12 Interlude: Gambler's ruin problem

12.1 Original gambler's ruin problem

Gambler has probability p of winning one unit and $q = 1 - p$ of loosing one unit.

What is the probability P_i that starting with i units, gambler's fortune will reach N before reaching 0 ?

We denote by X_n gambler's fortune at time n .

Possible states: $\{0\}, \{1\}, \dots, \{N\}$.

Process (X_n) = Markov chain

Transition probability $P_{k,l}$:

$$\begin{aligned}P_{0,0} &= 1 \\P_{N,N} &= 1 \\ \forall k \in \{1, \dots, N-1\} \quad P_{k,k+1} &= p \\ P_{k,k-1} &= q\end{aligned}$$

Conditionning on the outcome of the initial play

$$\begin{aligned}\forall i \in \{1, \dots, N-1\} \quad P_i &= p P_{i+1} + q P_{i-1} \\ P_0 &= 0 \\ P_N &= 1 \\ P_{i+1} - P_i &= \frac{q}{p} (P_i - P_{i-1})\end{aligned}$$

So,

$$P_i = \begin{cases} \frac{1 - (q/p)^i}{1 - (q/p)^N}, & \text{if } p \neq \frac{1}{2} \\ \frac{i}{N}, & \text{if } p = \frac{1}{2} \end{cases}$$

When $N \rightarrow \infty$,

$$P_i \rightarrow \begin{cases} 1 - (q/p)^i, & \text{if } p \neq \frac{1}{2} \\ 0, & \text{if } p = \frac{1}{2} \end{cases}$$

12.2 Another gambler's ruin problem

Competition

- Gambler against Banker.
- At the beginning, gambler's fortune = banker's fortune minus n units
- Gambler's fortune can be negative

Game takes end if gambler's fortune = banker's fortune at a certain time t .

What is the probability of success?

Note q_n this probability. We have: $q_0 = 1$ and $q_n \rightarrow 0$ when $n \rightarrow \infty$. Also by Markov's property,

$$q_n = q q_{n-1} + p q_{n+1} \quad (13)$$

Proposition 14. *We have $q_n = \left(\frac{q}{p}\right)^n$ when $n > 0$ and $q_n = 1$ when $n \leq 0$.*

[An Introduction to Probability Theory and Its Applications](#), W. Feller (1957)

13 Nakamoto's Analysis

13.1 Some definitions

Definition 15. Let $n \in \mathbb{Z}$. We denote by q_n the probability of the attacker \mathbf{A} to catch up honest miners whereas \mathbf{A} 's blockchain is n blocks behind.

Same problem as gambler's ruin problem!

$$q_n = \left(\frac{q}{p}\right)^n \quad (14)$$

Definition 16. For, $z \in \mathbb{N}$, the probability of success of a double-spending attack is denoted by $P(z)$.

Problem: $P(z) = ?$

Note 17. The probability $P(z)$ is evaluated at $t = 0$. The double-spending attack cannot be successful before $t = \mathbf{S}_z$.

13.2 Formula for $P(z)$

When $t = \mathbf{S}_z$, the attacker has mined $\mathbf{N}'(\mathbf{S}_z)$ blocks. By conditionning on $\mathbf{N}'(\mathbf{S}_z)$, we get:

$$\begin{aligned} P(z) &= \sum_{k=0}^{\infty} \mathbb{P}[\mathbf{N}'(\mathbf{S}_z) = k] q_{z-k} \\ &= \mathbb{P}[\mathbf{N}'(\mathbf{S}_z) \geq z] + \sum_{k=0}^{z-1} \mathbb{P}[\mathbf{N}'(\mathbf{S}_z) = k] q_{z-k} \\ &= 1 - \sum_{k=0}^{z-1} \mathbb{P}[\mathbf{N}'(\mathbf{S}_z) = k] \end{aligned}$$

$$\begin{aligned}
& + \sum_{k=0}^{z-1} \mathbb{P}[\mathbf{N}'(\mathbf{S}_z) = k] q_{z-k} \\
= & 1 - \sum_{k=0}^{z-1} \mathbb{P}[\mathbf{N}'(\mathbf{S}_z) = k] (1 - q_{z-k})
\end{aligned}$$

13.3 Satoshi's approximation

White paper, Section 11 **Calculations**

According to Satoshi,

$$\mathbf{S}_z \approx \mathbb{E}[\mathbf{S}_z]$$

and

$$\begin{aligned}
\mathbf{N}'(\mathbf{S}_z) & \approx \mathbf{N}'(\mathbb{E}[\mathbf{S}_z]) \\
& \approx \mathbf{N}'(z \cdot \mathbb{E}[\mathbf{T}]) \\
& \approx \mathbf{N}'\left(z \cdot \frac{\tau_0}{p}\right)
\end{aligned}$$

So, $\mathbf{N}'(\mathbf{S}_z) \approx$ Poisson process with parameter λ given by

$$\begin{aligned}
\lambda & = \alpha' \cdot z \cdot \frac{\tau_0}{p} \\
& = z \cdot \frac{q}{p}
\end{aligned}$$

The recipient waits until the transaction has been added to a block and z blocks have been linked after it. He doesn't know the exact amount of progress the attacker has made, but assuming the honest blocks took the average expected time per block, the attacker's potential progress will be a **Poisson distribution** with expected value:

$$\lambda = z \frac{q}{p}$$

Definition 18. We denote by $P_{\text{SN}}(z)$ the (false) formula obtained by Satoshi in Bitcoin's white paper.

Then,

$$P_{\text{SN}}(z) = 1 - \sum_{k=0}^{z-1} \frac{\lambda^k e^{-\lambda}}{k!} \left(1 - \left(\frac{q}{p} \right)^{z-k} \right) \quad (15)$$

Converting to C code...

```
#include <math.h>
double AttackerSuccessProbability(double q, int z)
{
    double p = 1.0 - q;
    double lambda = z * (q / p);
    double sum = 1.0;
    int i,k;
    for (k=0; k<=z; k++)
    {
        double poisson = exp(-lambda);
```

```

    for (i=1; i<=k; i++)
        poisson *= lambda/i;
    sum -= poisson * (1 - pow(q / p, z - k));
}
return sum;
}

```

However,

$$P(z) \neq P_{\text{SN}}(z)$$

since

$$\boldsymbol{N}'(\boldsymbol{S}_z) \neq \boldsymbol{N}'(\mathbb{E}[\boldsymbol{S}_z])$$

14 A correct analysis of double-spending attack

14.1 Meni Rosenfeld's correction

Set $\mathbf{X}_n := \mathbf{N}'(\mathbf{S}_n)$.

Proposition 19. *The random variable \mathbf{X}_n has a negative binomial distribution with parameters (n, p) , i.e., for $k \geq 0$*

$$\mathbb{P}[\mathbf{X}_n = k] = p^n q^k \binom{k+n-1}{k}$$

Proof. We have $\mathbf{S}_n \sim \Gamma(\alpha, n)$ i.e.,

$$f_{\mathbf{S}_n}(t) = \frac{\alpha^n}{(n-1)!} t^{n-1} e^{-\alpha t}$$

with $f_{\mathbf{S}_n}(t)$ = density of \mathbf{S}_n . So,

$$\begin{aligned} \mathbb{P}[\mathbf{X}_n = k] &= \int_0^{+\infty} \mathbb{P}[\mathbf{N}'(\mathbf{S}_n) = k | \mathbf{S}_n = t] f_{\mathbf{S}_n}(t) dt \\ &= \int_0^{+\infty} \frac{(\alpha' t)^k}{k!} e^{-\alpha' t} \frac{\alpha^n}{(n-1)!} t^{n-1} e^{-\alpha t} dt \\ &= \frac{p^n q^k}{(n-1)! k!} \int_0^{+\infty} t^{k+n-1} dt \\ &= \frac{p^n q^k}{(n-1)! k!} \cdot (k+n-1)! \end{aligned}$$

□

“The attacker’s potential progress” is not “a Poisson distribution with expected value $\lambda = z \frac{q}{p}$ ”...

Already remarked in 2012 (probably seen by Satoshi...)

Analysis of Hashrate-Based Double-Spending, Meni Rosenfeld, preprint, First Version December 11, 2012, p.7.

Proposition 20. *(Probability of success of the attacker) The probability of success of a double-spending attack is*

$$P(z) = 1 - \sum_{k=0}^{z-1} (p^z q^k - q^z p^k) \binom{k+z-1}{k}$$

Proof. Direct application of Section 13.2 and Proposition 19. □

14.2 Numerical Applications

For $q = 0.1$,

z	$P(z)$	$P_{\text{SN}}(z)$
0	1	1
1	0.2	0.2045873
2	0.0560000	0.0509779
3	0.0171200	0.0131722
4	0.0054560	0.0034552
5	0.0017818	0.0009137
6	0.0005914	0.0002428
7	0.0001986	0.0000647
8	0.0000673	0.0000173
9	0.0000229	0.0000046
10	0.0000079	0.0000012

For $q = 0.3$,

z	$P(z)$	$P_{\text{SN}}(z)$
0	1	1
5	0.1976173	0.1773523
10	0.0651067	0.0416605
15	0.0233077	0.0101008
20	0.0086739	0.0024804
25	0.0033027	0.0006132
30	0.0012769	0.0001522
35	0.0004991	0.0000379
40	0.0001967	0.0000095
45	0.0000780	0.0000024
50	0.0000311	0.0000006

Solving for P less than 0.1%:

q	z	z_{SN}
0.1	6	5
0.15	9	8
0.20	18	11
0.25	20	15
0.3	32	24
0.35	58	41
0.40	133	89

Satoshi underestimates $P(z)$...

15 A closed form formula

References.

[Hanbook of Mathematical Functions](#), M. Abramovitch, I.A. Stegun, Dover NY (1970).

[Digital Library of Mathematical Functions](http://dlmf.nist.gov),
<http://dlmf.nist.gov>

Definition 21. The *Gamma function* is defined for $x > 0$ by

$$\Gamma(x) := \int_0^{+\infty} t^{x-1} e^{-t} dt$$

The *incomplete Bêta function* is defined for $a, b > 0$ and $x \in [0, 1]$ by

$$B_x(a, b) := \int_0^x t^{a-1} (1-t)^{b-1} dt$$

The (classical) *Bêta function* is defined for $a, b > 0$ by

$$B(a, b) := B_1(a, b)$$

The *regularized Bêta function* is defined by

$$I_x(a, b) := \frac{B_x(a, b)}{B(a, b)}$$

Classical result: for $a, b > 0$,

$$B(a, b) = \frac{\Gamma(a) \Gamma(b)}{\Gamma(a+b)}$$

Theorem 22. *We have:*

$$P(z) = I_s(z, 1/2)$$

with $s = 4pq < 1$.

Proof. It turns out that the cumulative distribution function of a negative binomial random variable \mathbf{X} (same notation as above) is

$$\begin{aligned} F_{\mathbf{X}}(k) &= \mathbb{P}[\mathbf{X} \leq k] \\ &= 1 - I_p(k+1, z) \end{aligned}$$

By parts,

$$I_p(k, z) - I_p(k+1, z) = \frac{p^k q^z}{k B(k, z)}$$

So,

$$P(z) = 1 - I_p(z, z) + I_q(z, z)$$

Classical symmetry relation for Bêta function:

$$I_p(a, b) + I_q(b, a) = 1$$

(change of variable $t \mapsto 1 - t$ in the definition). So,

$$I_p(z, z) + I_q(z, z) = 1$$

We also use:

$$I_q(z, z) = \frac{1}{2} I_s(z, 1/2)$$

with $s = 4pq$. □

Classical function `pbeta` implemented in R gives the true double-spending attack success probability.

16 Asymptotic analysis

According to Satoshi,

Given our assumption that $p > q$, the probability drops **exponentially** as the number of blocks the attacker has to catch up with increases.

A result which has never been proven...

Lemma 23. *Let $f \in \mathcal{C}^1(\mathbb{R}_+)$ with $f(0) \neq 0$ and absolute convergent integral. Then,*

$$\int_0^{+\infty} f(u) e^{-zu} du \sim \frac{f(0)}{z}$$

Lemma 24. *For $b > 0$ and $s \in [0, 1]$, we have when $z \gg 1$,*

$$B_s(z, b) \sim \frac{s^z}{z} (1-s)^{b-1}$$

Proof. By the change of variable $u = \ln(s/t)$ in the definition of $B_s(z, b) = \int_0^s t^{z-1} (1-t)^{b-1} dt$,

$$B_s(z, b) = s^z \int_0^{+\infty} (1 - s e^{-u})^{b-1} e^{-zu} du$$

Then, we apply Lemma 23 with $f(u) := (1 - s e^{-u})^{b-1}$.

□

Proposition 25. *When $z \rightarrow \infty$, we have:*

$$P(z) \sim \frac{s^z}{\sqrt{\pi(1-s)} z}$$

with $s = 4pq < 1$.

Proof. By Stirling formula,

$$\begin{aligned} B(z, 1/2) &= \frac{\Gamma(z) \Gamma(1/2)}{\Gamma(z + 1/2)} \\ &\sim \sqrt{\frac{\pi}{z}} \end{aligned}$$

So,

$$\begin{aligned} P(z) &= I_s(z, 1/2) \\ &\sim \frac{(1-s)^{-\frac{1}{2}} \frac{s^z}{z}}{\sqrt{\frac{\pi}{z}}} \\ &\sim \frac{s^z}{\sqrt{\pi(1-s)} z} \end{aligned}$$

□

17 A more accurate risk analysis

The merchant waits for z blocks. Once it has been done, he knows how long it took... Denote this number by τ_1 . In average, it should take $\mathbb{E}[z \mathbf{T}] = \frac{z \tau_0}{p}$.

Definition 26. Set $\kappa := \frac{p \tau_1}{z \tau_0}$

Dimensionless parameter.

Satoshi's approximation: $\kappa = 1$...

Instead of computing $P(z)$, let us compute $P(z, \kappa)$.

Probability for a successful double-spending attack knowing that z blocks have been mined by the honest miners at $\mathbf{S}_z = \tau_1$.

Note 27. We have $P_{\text{SN}}(z) = P(z, 1)$.

Note 28. Two different probabilities.

- Theoretical probability $P(z)$ calculated at $T = 0$ by the attacker or the merchant.
- concrete probability $P(z, \kappa)$ calculated at $T = \tau_1$ by the merchant .

Number of bocks mined by the attacker at $T = \tau_1$ unknown to the merchant = Poisson distribution parameter $\lambda(z, \kappa)$:

$$\begin{aligned}\lambda(z, \kappa) &= \alpha' \tau_1 \\ &= \frac{q}{\tau_0} \cdot \frac{z \kappa \tau_0}{p} \\ &= \frac{z q}{p} \kappa\end{aligned}$$

i.e.,

$$\mathbb{P}[\mathbf{N}'(\tau_1) = k] = \frac{\left(\frac{z q}{p} \kappa\right)^k}{k!} e^{-\frac{z q}{p} \kappa}$$

Definition 29. *The regularized Gamma function is defined by:*

$$\Gamma(s, x) := \int_x^{+\infty} t^{s-1} e^{-t} dt$$

The regularized incomplete Gamma function is:

$$Q(s, x) := \frac{\Gamma(s, x)}{\Gamma(s)}$$

It turns out that

$$Q(z, \lambda) = \sum_{k=0}^{z-1} \frac{\lambda^k}{k!} e^{-\lambda}$$

So,

Theorem 30. *We have:*

$$P(z, \kappa) = 1 - Q\left(z, \frac{\kappa z q}{p}\right) + \left(\frac{q}{p}\right)^z e^{\kappa z \frac{p-q}{p}} Q(z, \kappa z)$$

Proof. We have:

$$\begin{aligned} P(z, \kappa) &= \mathbb{P}[\mathbf{N}'(\tau_1) \geq z] + \sum_{k=0}^{z-1} \mathbb{P}[\mathbf{N}'(\tau_1) = k] q_{z-k} \\ &= 1 - \sum_{k=0}^{z-1} \frac{\lambda(z, \kappa)^k}{k!} e^{-\lambda(z, \kappa)} \\ &\quad + \sum_{k=0}^{z-1} \left(\frac{q}{p}\right)^{z-k} \cdot \frac{\lambda(z, \kappa)^k}{k!} e^{-\lambda(z, \kappa)} \\ &= 1 - Q\left(z, \frac{\kappa z q}{p}\right) + \left(\frac{q}{p}\right)^z e^{\kappa z \frac{p-q}{p}} Q(z, \kappa z) \end{aligned}$$

□

18 Asymptotics Analysis

Lemma 31. *We have:*

i. For $\mu \in]0, 1[$, $Q(z, \mu z) \rightarrow 1$ and

$$1 - Q(z, \mu z) \sim \frac{1}{1 - \mu} \frac{1}{\sqrt{2\pi z}} e^{-z(\mu - 1 - \ln \mu)}$$

ii. For $\mu = 1$, $Q(z, z) \rightarrow \frac{1}{2}$ and

$$\frac{1}{2} - Q(z, z) \sim \frac{1}{3\sqrt{2\pi z}}$$

iii. For $\mu \in]1, +\infty[$,

$$Q(z, \mu z) \sim \frac{1}{\mu - 1} \frac{1}{\sqrt{2\pi z}} e^{-z(\mu - 1 - \ln \mu)}$$

Proposition 32. We have $P_{\text{SN}}(z) \sim \frac{e^{-zc\left(\frac{q}{p}\right)}}{2}$ with

$$c(\mu) := \mu - 1 - \ln \mu$$

Proof. It follows that

$$\begin{aligned} 1 - Q\left(z, \frac{q}{p} z\right) &\sim \frac{1}{1 - \frac{q}{p}} \frac{1}{\sqrt{2\pi z}} e^{-zc\left(\frac{q}{p}\right)} \\ \left(\frac{q}{p}\right)^z e^{\kappa z \frac{p-q}{p}} Q(z, z) &\sim \frac{1}{2} e^{-zc\left(\frac{q}{p}\right)} \end{aligned}$$

□

More generally, we have **5 different regimes**.

Proposition 33. When $z \rightarrow +\infty$, we have:

- For $0 < \kappa < 1$, $P(z, \kappa) \sim \frac{1}{1 - \kappa \frac{q}{p}} \frac{1}{\sqrt{2\pi z}} e^{-zc\left(\kappa \frac{q}{p}\right)}$
- For $\kappa = 1$, $P(z, 1) = P_{\text{SN}}(z) \sim \frac{e^{-zc\left(\frac{q}{p}\right)}}{2}$
- For $1 < \kappa < \frac{p}{q}$,

$$P(z, \kappa) \sim \frac{\kappa \left(1 - \frac{q}{p}\right)}{(\kappa - 1) \left(1 - \kappa \frac{q}{p}\right)} \frac{1}{\sqrt{2\pi z}} e^{-zc\left(\kappa \frac{q}{p}\right)}$$

- For $\kappa = \frac{p}{q}$, $P\left(z, \frac{p}{q}\right) \rightarrow \frac{1}{2}$ and

$$P\left(z, \frac{p}{q}\right) - \frac{1}{2} \sim \frac{1}{\sqrt{2\pi z}} \left(\frac{1}{3} + \frac{q}{p-q} \right)$$

- For $\kappa > \frac{p}{q}$, $P(z, \kappa) \rightarrow 1$ and

$$1 - P(z, \kappa) \sim \frac{\kappa \left(1 - \frac{q}{p}\right)}{\left(\kappa \frac{q}{p} - 1\right) (\kappa - 1)} \frac{1}{\sqrt{2\pi z}} e^{-zc\left(\kappa \frac{q}{p}\right)}$$

Proof. Repetitive application of Lemma 31. □

19 Comparison between $P(z)$ and $P_{\text{SN}}(z)$

19.1 Asymptotic behaviours

The asymptotic behaviours of $P(z)$ and $P_{\text{SN}}(z)$ are quite different

Proposition 34. *We have $P_{\text{SN}}(z) \prec P(z)$*

Proposition 35. *We have:*

$$\frac{q}{p} - 1 - \ln \frac{q}{p} - \ln \left(\frac{1}{4pq} \right) = \ln 4 - 2 + x - 2 \ln x$$

with $x = \frac{1}{p} \in [1, 2]$.

19.2 Bounds for $P(z)$ and $P_{\text{SN}}(z)$

Goal: compute an explicit rank z_0 such that

$$P_{\text{SN}}(z) < P(z)$$

for all $z > z_0$.

19.2.1 Upper and lower bounds for $P(z)$

Remember that $s = 4pq$.

We'll use [Gautschi's inequalities](#).

Proposition 36. *For any $z > 1$,*

$$\sqrt{\frac{z}{z+1}} \frac{s^z}{\sqrt{\pi z}} \leq P(z) \leq \frac{s^z}{\sqrt{\pi(1-s)z}}$$

Proof. The function $x \mapsto (1-x)^{-\frac{1}{2}}$ is non-decreasing. So, by definition of I_s ,

$$\begin{aligned} P(z) = I_s\left(z, \frac{1}{2}\right) &= \frac{\Gamma\left(z + \frac{1}{2}\right)}{\Gamma\left(\frac{1}{2}\right) \Gamma(z)} \int_0^s t^{z-1} (1-t)^{-\frac{1}{2}} dt \\ &\leq \frac{\Gamma\left(z + \frac{1}{2}\right)}{\Gamma\left(\frac{1}{2}\right) \Gamma(z)} \int_0^s t^{z-1} (1-s)^{-\frac{1}{2}} dt \\ &\leq \left(\frac{\Gamma\left(z + \frac{1}{2}\right)}{\sqrt{z} \Gamma(z)} \right) \frac{s^z}{\sqrt{\pi(1-s)z}} \end{aligned}$$

Then, we use Gautschi's inequality

$$\frac{\Gamma\left(z + \frac{1}{2}\right)}{\sqrt{z} \Gamma(z)} < 1$$

On the same way, using other side of Gautschi's inequality,

$$\begin{aligned}
P(z) = I_s\left(z, \frac{1}{2}\right) &\geq \frac{\Gamma\left(z + \frac{1}{2}\right)}{\Gamma\left(\frac{1}{2}\right) \Gamma(z)} \int_0^s t^{z-1} dt \\
&\geq \frac{\Gamma\left(z + \frac{1}{2}\right)}{\Gamma\left(\frac{1}{2}\right) \Gamma(z)} \frac{s^z}{\sqrt{\pi z}} \\
&\geq \sqrt{\frac{z}{z+1}} \frac{s^z}{\sqrt{\pi z}}
\end{aligned}$$

□

19.2.2 An upper bound for $P_{\text{SN}}(z)$

Lemma 37. *Let $z \in \mathbb{N}^*$ and $\lambda \in \mathbb{R}_+^*$. We have:*

i. If $\lambda \in]0, 1[$, then

$$1 - Q(z, \lambda z) < \frac{1}{1 - \lambda} \frac{1}{\sqrt{2\pi z}} e^{-z(\lambda - 1 - \ln \lambda)}$$

ii. If $\lambda = 1$, $Q(z, z) < \frac{1}{2}$.

Proof. Let us prove i. first.

We use dlmf.nist.gov/8.7.1,

$$\gamma(a, x) = e^{-x} x^a \sum_{n=0}^{+\infty} \frac{\Gamma(a)}{\Gamma(a + n + 1)} x^n$$

valid for $a, x \in \mathbb{R}$ and

$$\begin{aligned}
\gamma(a, x) &:= \int_x^{+\infty} t^{a-1} e^{-t} dt \\
&= \Gamma(a) - \Gamma(a, x)
\end{aligned}$$

Let $\lambda \in]0, 1[$. Using recursively $\Gamma(z+1) = z \Gamma(z)$,

$$\begin{aligned}
\gamma(z, z\lambda) &= e^{-z\lambda} (z\lambda)^z \sum_{n=0}^{+\infty} \frac{\Gamma(z)}{\Gamma(z+n+1)} (z\lambda)^n \\
&= \frac{\lambda^z z^{z-1} e^{-z\lambda}}{1-\lambda} z(1-\lambda) \cdot \\
&\quad \cdot \left(\frac{1}{z} + \frac{1}{z(z+1)} (z\lambda) + \dots \right) \\
&\leq \frac{\lambda^z z^{z-1} e^{-z\lambda}}{1-\lambda} z(1-\lambda) \cdot \\
&\quad \cdot \left(\frac{1}{z} + \frac{1}{z^2} (z\lambda) + \frac{1}{z^3} (z\lambda)^2 + \dots \right) \\
&\leq \frac{\lambda^z z^{z-1} e^{-z\lambda}}{1-\lambda} z(1-\lambda) \cdot \frac{1}{z} \cdot (1 + \lambda + \dots) \\
&\leq \frac{\lambda^z z^{z-1} e^{-z\lambda}}{1-\lambda} z(1-\lambda) \cdot \frac{1}{z} \cdot \frac{1}{1-\lambda} \\
&\leq \frac{\lambda^z z^{z-1} e^{-z\lambda}}{1-\lambda}
\end{aligned}$$

By dlmf.nist.gov/5.6.1,

$$\frac{1}{\Gamma(z)} < \frac{e^z}{\sqrt{2\pi} z^{z-1}}$$

So, for $0 < \lambda < 1$,

$$\begin{aligned}
1 - Q(z, \lambda z) &= \frac{\gamma(z, z\lambda)}{\Gamma(z)} \\
&< \frac{1}{1-\lambda} \frac{1}{\sqrt{2\pi} z} e^{-z(\lambda-1-\ln\lambda)}
\end{aligned}$$

The second inequality (ii) comes directly from dlmf.nist.gov/8.10.13 □

Proposition 38. *We have*

$$P_{\text{SN}}(z) < \frac{1}{1 - \frac{q}{p}} \frac{1}{\sqrt{2\pi}z} e^{-zc\left(\frac{q}{p}\right)} + \frac{1}{2} e^{-zc\left(\frac{q}{p}\right)}$$

with $c(\lambda) := \lambda - 1 - \ln \lambda$.

19.3 An explicit rank z_0

Lemma 39. *For $\mu, \psi, x > 0$, the inequality*

$$e^{-\psi x} < \frac{\mu}{\sqrt{x+1}}$$

is satisfied if $x > \frac{1}{2\sqrt{2}} - \frac{1+\sqrt{2}}{2\sqrt{2}} \frac{\ln(2\psi\mu^2)}{\psi}$.

Theorem 40. *Let $z \in \mathbb{N}^*$. A sufficient condition to get $P(z) < P_{\text{SN}}(z)$ is $z > z_0$ with*

$$z_0 := \text{Max} \left(\frac{2}{\pi \left(1 - \frac{q}{p}\right)^2}, \frac{1}{2\sqrt{2}} - \frac{1 + \frac{1}{\sqrt{2}} \ln\left(\frac{2\psi_0}{\pi}\right)}{2\psi_0} \right)$$

with

$$\psi_0 := \frac{q}{p} - 1 - \ln\left(\frac{q}{p}\right) - \ln\left(\frac{1}{4pq}\right) > 0$$

20 Securing Fast Payments

[On the Scalability and Security of Bitcoin](#), C. Decker, 2016. Chapter 8.

Group Thesis T. Bamert, L. Elsen, S. Welten, R. Wattenhofer, ETH Zurich.

[Have a snack, pay with Bitcoins](#), 2014 ?

Tradeoff between transaction speed and confirmation reliability in the Bitcoin network.

C. Decker and R. Wattenhofer, [Information propagation in the bitcoin network](#), 2013.

Two transactions from the same output T_A and T_V .

The attacker attempts to convince the merchant about the validity of T_V while broadcasting T_A to the network at the same time.

Goal: Hide T_A to the merchant but T_A must be included in a block of the blockchain

Influence of node sample size. Double spending-attack

20.1 Risk of information eclipsing

If the merchant forwards T_V to its neighboring nodes, they will verify and tentatively commit it to the local ledger. Should they later receive T_A , it will not be considered valid as it conflicts with T_V , and it will not be forwarded to the merchant. The merchant inadvertently shields itself against conflicting transactions like T_A , and will be unaware of the double-spending attempt.

20.2 Countermeasures

- The merchant should connect to a sufficiently large random sample of nodes in the Bitcoin network.
- The merchant should not accept incoming connections.
- The merchant can effectively avoid isolation by not relaying transaction T_V

As soon as a single node is uninfluenced by the attacker, it will forward T_A to the merchant, thus informing the merchant of the attempted double-spend

Many simulations: 192 200 000

At 100 nodes the merchant will not learn of a double-spending attempt in only 0.77% of all attempted double-spends.

Time before detection.

The time until the merchant detects the double-spending attack quickly decreases for larger sample sizes. The 99 percentile is at 6,29 seconds for 100 peers.

Transaction T_V should be seen first but not confirmed by the blockchain

Conclusion of the study

Bitcoin can be used as a reliable alternative for fast cashless payments.

But not scalable...

21 What is the cost of a double-spending attack?

Economic evaluation

21.1 Cost of mining

Mining during t with hashing power h has a cost C (for honest miners) which is proportionnal to t and h :
 $\exists \lambda > 0$ such that

$$C(h, t) = \lambda h t$$

Let B be the block reward. Today, $B = 12,5$ BTC
Parameter λ is adjusted so that

$$C(h + h', \tau_0) = B$$

Therefore,

$$\lambda (h + h') \tau_0 = B$$

and

$$\begin{aligned} C(h, t) &= \frac{h t}{(h + h') \tau_0} B \\ &= \frac{p t}{\tau_0} B \end{aligned}$$

Similarly, for an attacker,

$$C(h, t) = \frac{q t}{\tau_0} B$$

21.2 Classical double spending attack

Competition attacker/honest miners

Cost is a random variable

Cost function at $T = 0$

$$C = \frac{q \tau}{\tau_0} B$$

where τ is the stopping time:

$$\tau := \text{Inf} \{t \geq S_z / N'(t) \geq N(t)\}$$

Economic evaluation:

$$\begin{aligned} C &= \mathbb{E} \left[\frac{q \tau}{\tau_0} B \right] \\ &= \frac{q B}{\tau_0} \mathbb{E}[\tau] \\ &= +\infty \end{aligned}$$

Other possible stopping time:

$$\tau_T := \text{Inf} \{t \geq \mathbf{S}_z / \mathbf{N}'(t) \geq \mathbf{N}(t)\} \wedge T$$

Andresen & al:

We assume that the attacker will stop mining when he reaches $z + 1$ blocks on the fraudulent branch or when the honest miners reach $z + 1$ blocks on the main branch, whichever happens first.

$$\tilde{\tau} := \mathbf{S}_{z+1} \wedge \mathbf{S}'_{z+1}$$

Economic evaluation:

$$\begin{aligned} C &= \mathbb{E} \left[\frac{q \tilde{\tau}}{\tau_0} B \right] \\ &= \frac{q B}{\tau_0} \mathbb{E} [\mathbf{S}_{z+1} \wedge \mathbf{S}'_{z+1}] \end{aligned}$$

and \mathbf{S}_{z+1} and \mathbf{S}'_{z+1} are two independent random variables that has a Gamma distribution

21.3 Double spending attack and eclips attack

It is simply the cost for mining z blocks

$$C = \frac{q \tau}{\tau_0} B$$

with

$$\tau := \mathbf{S}'_z$$

With a deadline T :

$$C = \frac{q \tau}{\tau_0} B$$

with

$$\tau = S_z \wedge T$$

See Andresen & al...

The security of a transaction increases roughly logarithmically with the number of confirmations that it receives, where an attacker benefits from the increasing goods at risk but is also throttled by the increasing proof of work required. Additionally, we have demonstrated that, if merchants impose a conservative confirmation deadline, the eclipse attack does not increase an attacker's profit when his share of the mining power is less than 35% or more than 10 confirmations are required.