On profitability of Selfish Mining
(joint work with C. Grunspan)

Ricardo Pérez-Marco (CNRS, IMJ-PRG, Paris 7)

Paris Cryptofinance Seminar
Univ. Paris-Diderot, Paris

June 21, 2018

On profitability of Selfish Mining

1. Deviant mining strategies
2. On profitability
3. Martingale analysis
4. Lead-Stubborn Mining
5. Equal Fork Stubborn Mining
6. Attack on difficulty adjustment
7. Profitability after a difficulty adjustment
On profitability of Selfish Mining

1. Deviant mining strategies
2. On profitability
3. Martingale analysis
4. Lead-Stubborn Mining
5. Equal Fork Stubborn Mining
6. Attack on difficulty adjustment
7. Profitability after a difficulty adjustment

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Selfish Mining

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Selfish Mining

History
Selfish Mining

History

Selfish Mining

History

Selfish Mining

History

Selfish Mining

History


R. Pérez-Marco
On profitability of Selfish Mining (joint work with C. Grunspan)
Selfish Mining

History


■ Further papers and textbooks.

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Selfish Mining

History

- Further papers and textbooks.
Selfish Mining

History

- Further papers and textbooks.

Bitcoin Protocol rule “Bitcoin miners release blocks as soon as they are validated”.

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Selfish Mining

History

- Further papers and textbooks.

**Bitcoin Protocol rule** “Bitcoin miners release blocks as soon as they are validated”.

**Bitcoin Stability Conjecture** Protocol rules are aligned with self-interest of the network actors.
Description

R. Pérez-Marco
On profitability of Selfish Mining (joint work with C. Grunspan)
Description

General block withholding strategies

Withheld blocks trying to build an advantage with a relative hashing power \(0 < q < 1/2\). Timely release blocks to invalidate blocks validated by honest miners. Relies on a good connection so that a share of \(0 < \gamma \leq 1\) miners adopt the selfish block in case of competition.

Consequences

Slows the network, hence it reduces the total "Profit and Loss" (PnL) per unit time of the network. Creates a large amount of orphan blocks (hence, the attack is noticeable). All other things being equal, after 2016 blocks, the difficulty adjusts down.

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
General block withholding strategies

- Witheld blocks trying to build an advantage with a relative hashing power $0 < q < 1/2$. 
Description

General block withholding strategies

- Witheld blocks trying to build an advantage with a relative hashing power $0 < q < 1/2$.
- Timely release blocks to invalidate blocks validated by honest miners.
Description

General block withholding strategies

- Witheld blocks trying to build an advantage with a relative hashing power $0 < q < 1/2$.
- Timely release blocks to invalidate blocks validated by honest miners.
- Relies on a good connection so that a share of $0 < \gamma \leq 1$ miners adopt the selfish block in case of competition.
Description

General block withholding strategies

- Witheld blocks trying to build an advantage with a relative hashing power $0 < q < 1/2$.
- Timely release blocks to invalidate blocks validated by honest miners.
- Relies on a good connection so that a share of $0 < \gamma \leq 1$ miners adopt the selfish block in case of competition.
Description

General block withholding strategies

- Witheld blocks trying to build an advantage with a relative hashing power $0 < q < 1/2$.
- Timely release blocks to invalidate blocks validated by honest miners.
- Relies on a good connection so that a share of $0 < \gamma \leq 1$ miners adopt the selfish block in case of competition.

Consequences

- Slows the network, hence it reduces the total “Profit and Loss” (PnL) per unit if time of the network.
Description

General block withholding strategies

- Witheld blocks trying to build an advantage with a relative hashing power \( 0 < q < 1/2 \).
- Timely release blocks to invalidate blocks validated by honest miners.
- Relies on a good connection so that a share of \( 0 < \gamma \leq 1 \) miners adopt the selfish block in case of competition.

Consequences

- Slows the network, hence it reduces the total “Profit and Loss” (PnL) per unit if time of the network.
- Creates a large amount of orphan blocks (hence, the attack is noticeable)
Description

General block withholding strategies

- Withheld blocks trying to build an advantage with a relative hashing power $0 < q < 1/2$.
- Timely release blocks to invalidate blocks validated by honest miners.
- Relies on a good connection so that a share of $0 < \gamma \leq 1$ miners adopt the selfish block in case of competition.

Consequences

- Slows the network, hence it reduces the total “Profit and Loss” (PnL) per unit if time of the network.
- Creates a large amount of orphan blocks (hence, the attack is noticeable)
- All other things being equal, after 2016 blocks, the difficulty adjusts down.
Selfish mining algorithm

Let $\Delta \geq 0$ be the advance of the secret fork over the public blockchain. When the honest miners validate a block then (Selfish Mining (SM) algorithm):

- If $\Delta = 0$ the SM mines normally.
- If $\Delta = 1$ then the SM broadcasts his block. A competition follows.
- If $\Delta = 2$ then the SM broadcasts his secret fork.
- If $\Delta \geq 3$ then the SM broadcasts blocks from his secret fork to match the length of the public blockchain.

Except in the first two cases, the SM keeps working on top of his secret fork.
Selfish mining algorithm

Let $\Delta \geq 0$ be the advance of the secret fork over the public blockchain. When the honest miners validate a block then (Selfish Mining (SM) algorithm):

- If $\Delta = 0$ the SM mines normally.
- If $\Delta = 1$ then the SM broadcasts his block. A competition follows.
- If $\Delta = 2$ then the SM broadcasts his secret fork.
- If $\Delta \geq 3$ then the SM broadcasts blocks from his secret fork to match the length of the public blockchain.
- Except in the first two cases, the SM keeps working on top of his secret fork.
Selfish mining algorithm

Let $\Delta \geq 0$ be the advance of the secret fork over the public blockchain. When the honest miners validate a block then (Selfish Mining (SM) algorithm):

- If $\Delta = 0$ the SM mines normally.
Selfish mining algorithm

Let $\Delta \geq 0$ be the advance of the secret fork over the public blockchain. When the honest miners validate a block then (Selfish Mining (SM) algorithm):

- If $\Delta = 0$ the SM mines normally.
- If $\Delta = 1$ then the SM broadcasts his block. A competition follows.
- If $\Delta \geq 3$ then the SM broadcasts blocks from his secret fork to match the length of the public blockchain.

Except in the first two cases, the SM keeps working on top of his secret fork.
Selfish mining algorithm

Let $\Delta \geq 0$ be the advance of the secret fork over the public blockchain. When the honest miners validate a block then (Selfish Mining (SM) algorithm):

- If $\Delta = 0$ the SM mines normally.
- If $\Delta = 1$ then the SM broadcasts his block. A competition follows.
- If $\Delta = 2$ then the SM broadcasts his secret fork.
Let $\Delta \geq 0$ be the advance of the secret fork over the public blockchain. When the honest miners validate a block then (Selfish Mining (SM) algorithm):

- If $\Delta = 0$ the SM mines normally.
- If $\Delta = 1$ then the SM broadcasts his block. A competition follows.
- If $\Delta = 2$ then the SM broadcasts his secret fork.
- If $\Delta \geq 3$ then the SM broadcasts blocks from his secret fork to match the length of the public blockchain.
Selfish mining algorithm

Let $\Delta \geq 0$ be the advance of the secret fork over the public blockchain. When the honest miners validate a block then (Selfish Mining (SM) algorithm):

- If $\Delta = 0$ the SM mines normally.
- If $\Delta = 1$ then the SM broadcasts his block. A competition follows.
- If $\Delta = 2$ then the SM broadcasts his secret fork.
- If $\Delta \geq 3$ then the SM broadcasts blocks from his secret fork to match the length of the public blockchain.
- Except in the first two cases, the SM keeps working on top of his secret fork.
Stubborn Mining algorithms


Lead-Stubborn Mining (LStM) When \( \Delta \geq 2 \) as in SM with \( \Delta \geq 3 \), and for \( \Delta = 1 \) releases all the secret fork and mines normally on top of it.

Equal Fork Stubborn Mining (EFStM) As in the previous case for \( \Delta = 1 \), but if the deviant miner finds a new block he does not reveal it.

Other "Trail Mining" strategies would be discussed elsewhere in the general context of Catch-up Mining (CM).

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Stubborn Mining algorithms

Stubborn Mining algorithms


These are variations of SM algorithm.
Stubborn Mining algorithms


These are variations of SM algorithm.

- Lead-Stubborn Mining (LStM) When $\Delta \geq 2$ as in SM with $\Delta \geq 3$, and for $\Delta = 1$ releases all the secret fork and mines normally on top of it.
Stubborn Mining algorithms


These are variations of SM algorithm.

- **Lead-Stubborn Mining (LStM)** When $\Delta \geq 2$ as in SM with $\Delta \geq 3$, and for $\Delta = 1$ releases all the secret fork and mines normally on top of it.

- **Equal Fork Stubborn Mining (EFStM)** As in the previous case for $\Delta = 1$, but if the deviant miner finds a new block he does not reveal it.
Stubborn Mining algorithms


These are variations of SM algorithm.

- **Lead-Stubborn Mining (LStM)** When $\Delta \geq 2$ as in SM with $\Delta \geq 3$, and for $\Delta = 1$ releases all the secret fork and mines normally on top of it.

- **Equal Fork Stubborn Mining (EFStM)** As in the previous case for $\Delta = 1$, but if the deviant miner finds a new block he does not reveal it.
Stubborn Mining algorithms


These are variations of SM algorithm.

- **Lead-Stubborn Mining (LStM)** When $\Delta \geq 2$ as in SM with $\Delta \geq 3$, and for $\Delta = 1$ releases all the secret fork and mines normally on top of it.

- **Equal Fork Stubborn Mining (EFStM)** As in the previous case for $\Delta = 1$, but if the deviant miner finds a new block he does not reveal it.

Other “Trail Mining” strategies would be discussed elsewhere in the general context of Catch-up Mining (CM).
Markov chain model
Markov chain model

Markov chain or “state machine” models.
Markov chain model

Markov chain or “state machine” models.

For SM
Markov chain model

Markov chain or “state machine” models.

For SM

\[ \begin{align*}
0 & \xrightarrow{(1-\gamma)\beta} 1 \\
0 & \xrightarrow{\alpha} 2 \\
0 & \xrightarrow{\beta} 3
\end{align*} \]
Markov chain model

Markov chain or “state machine” models.

For SM

For LStM
Markov chain model

Markov chain or “state machine” models.

For SM

![Markov chain diagram for SM]

For LStM

![Markov chain diagram for LStM]
Markov model limitations

The profitability analysis depends in a fundamental way on the duration of the attack cycles. The stationnary probability of the Markov model computes the probability of being in a given state in a steady regime. The Markov model offers no insight of the duration of the attack cycles nor on the time to reach a steady regime. There is no proper analysis of profitability in the literature.
Markov model limitations

- The profitability analysis depends in a fundamental way on the duration of the attack cycles.
Markov model limitations

- The profitability analysis depends in a fundamental way on the duration of the attack cycles.
- The stationnary probability of the Markov model computes the probability of being in a given state in a steady regime.
Markov model limitations

- The profitability analysis depends in a fundamental way on the **duration of the attack cycles**.
- The **stationary probability** of the Markov model computes the probability of being in a given state in a steady regime.
- The Markov model offers **no insight** of the duration of the attack cycles nor on the time to reach a steady regime.
Markov model limitations

- The profitability analysis depends in a fundamental way on the duration of the attack cycles.
- The stationnary probability of the Markov model computes the probability of being in a given state in a steady regime.
- The Markov model offers no insight of the duration of the attack cycles nor on the time to reach a steady regime.
- There is no proper analysis of profitability in the literature.
Profit and Loss

• Profit and Loss (PnL) of a business:
  \[ PnL = R - C = \text{Profit} - \text{Cost} \]

• What counts is Profit and Loss per unit time (PnLt):
  \[ PnLt = R_t - C_t = (\text{Profit per unit time}) - (\text{Cost per unit time}) \]

• Key observation: \( C_t \) for a non-stopping mining operation is the same for honest mining or a deviant strategy.

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Profit and Loss

- Profit and Loss (PnL) of a business

\[ \text{PnL} = R - C = \text{Profit} - \text{Cost} \]
Profit and Loss

- **Profit and Loss (PnL) of a business**
  \[ PnL = R - C = \text{Profit} - \text{Cost} \]

- What counts is **Profit and Loss per unit time (PnLt)**
  \[ PnLt = R_t - C_t = (\text{Profit per unit time}) - (\text{Cost per unit time}) \]
Profit and Loss

- **Profit and Loss (PnL) of a business**
  \[ PnL = R - C = \text{Profit} - \text{Cost} \]

- **What counts is Profit and Loss per unit time (PnLt)**
  \[ PnLt = R_t - C_t = (\text{Profit per unit time}) - (\text{Cost per unit time}) \]

- **Key observation**: \( C_t \) for a non-stopping mining operation is the same for honest mining or a deviant strategy.
Profit and Loss

- **Profit and Loss (PnL) of a business**

\[ PnL = R - C = \text{Profit} - \text{Cost} \]

- **What counts is Profit and Loss per unit time (PnLt)**

\[ PnLt = R_t - C_t = (\text{Profit per unit time}) - (\text{Cost per unit time}) \]

- **Key observation**: \( C_t \) for a non-stopping mining operation is the same for honest mining or a deviant strategy.
Repetition games
Repetition games

Definition (Repetition games)

A repetition game follows an strategy of repeated cycles.
Repetition games

Definition (Repetition games)

A repetition game follows an strategy of repeated cycles.
Repetition games

Definition (Repetition games)

A repetition game follows an strategy of repeated cycles. Let $R$, $C$ and $T$ be random variables resp. of revenue, cost and duration over a cycle.
Repetition games

Definition (Repetition games)

A repetition game follows an strategy of repeated cycles. Let $R$, $C$ and $T$ be random variables resp. of revenue, cost and duration over a cycle. The game is integrable when

$$\mathbb{E}[T] < +\infty.$$
Repetition games

Definition (Repetition games)

A repetition game follows an strategy of repeated cycles. Let $R$, $C$ and $T$ be random variables resp. of revenue, cost and duration over a cycle. The game is integrable when

$$\mathbb{E}[T] < +\infty.$$ 

Theorem (Profitability of integrable games)

$$\mathbb{E}[PnLt] = \frac{\mathbb{E}[R] - \mathbb{E}[C]}{\mathbb{E}[T]}.$$ 

R. Pérez-Marco

On profitability of Selfish Mining  (joint work with C. Grunspan)
Proof of the Profitability Theorem

Proof.

$R_i$, $C_i$ and $T_i$ values for the $i$-cycle.
Proof of the Profitability Theorem

Proof.

$R_i$, $C_i$ and $T_i$ values for the $i$-cycle.
Proof of the Profitability Theorem

Proof.

$R_i$, $C_i$ and $T_i$ values for the $i$-cycle. The $(R_i)$ (resp. $(C_i)$, $(T_i)$) are integrable i.i.d. random variables.
Proof of the Profitability Theorem

Proof.

$R_i$, $C_i$ and $T_i$ values for the $i$-cycle. The $(R_i)$ (resp. $(C_i)$, $(T_i)$) are integrable i.i.d. random variables. Let $P_{nLt}$ be the $P_{nLt}$ after $n$ cycles:
Proof of the Profitability Theorem

Proof.

$R_i$, $C_i$ and $T_i$ values for the $i$-cycle. The $(R_i)$ (resp. $(C_i)$, $(T_i)$) are integrable i.i.d. random variables. Let $P_n L_t_n$ be the $P_n L_t$ after $n$ cycles:

$$P_n L_{t_n} = \sum_{i=1}^{n} \frac{R_i - \sum_{i=1}^{n} C_i}{\sum_{i=1}^{n} T_i} = \frac{1}{n} \sum_{i=1}^{n} R_i - \frac{1}{n} \sum_{i=1}^{n} C_i \quad \frac{1}{n} \sum_{i=1}^{n} T_i$$
Proof of the Profitability Theorem

Proof.

$R_i$, $C_i$ and $T_i$ values for the $i$-cycle. The $(R_i)$ (resp. $(C_i)$, $(T_i)$) are integrable i.i.d. random variables. Let $PnLt_n$ be the $PnLt$ after $n$ cycles:

$$PnLt_n = \frac{\sum_{i=1}^{n} R_i - \sum_{i=1}^{n} C_i}{\sum_{i=1}^{n} T_i} = \frac{1}{n} \sum_{i=1}^{n} R_i - \frac{1}{n} \sum_{i=1}^{n} C_i.$$

By the Strong Law of Large Numbers we have that almost surely
Proof.

$R_i$, $C_i$ and $T_i$ values for the $i$-cycle. The $(R_i)$ (resp. $(C_i)$, $(T_i)$) are integrable i.i.d. random variables. Let $PnLt_n$ be the $PnLt$ after $n$ cycles:

$$PnLt_n = \frac{\sum_{i=1}^{n} R_i - \sum_{i=1}^{n} C_i}{\sum_{i=1}^{n} T_i} = \frac{1}{n} \sum_{i=1}^{n} R_i - \frac{1}{n} \sum_{i=1}^{n} C_i.$$

By the Strong Law of Large Numbers we have that almost surely

$$\mathbb{E}[PnLt] = \lim_{n \to +\infty} PnL_n = \frac{\mathbb{E}[R] - \mathbb{E}[C]}{\mathbb{E}[T]}.$$
Revenue ratio

To compare integrable games with repetition and equal cost the benchmark is the Revenue Ratio.

Definition (Revenue Ratio)

The revenue ratio of a game with repetition is

\[ P = \frac{\mathbb{E}[R]}{\mathbb{E}[T]} \]

Corollary

Let \( S_1 \) and \( S_2 \) be integrable non-stopping mining strategies. Strategy \( S_1 \) is more profitable than strategy \( S_2 \) if and only if

\[ P(S_1) \geq P(S_2) \]
Revenue ratio

To compare integrable games with repetition and equal cost the benchmark is the Revenue Ratio.
Revenue ratio

To compare integrable games with repetition and equal cost the benchmark is the Revenue Ratio

Definition (Revenue Ratio)

The revenue ratio of a game with repetition is

\[ P = \frac{\mathbb{E}[R]}{\mathbb{E}[T]} \]
Revenue ratio

To compare integrable games with repetition and equal cost the benchmark is the Revenue Ratio

Definition (Revenue Ratio)

The revenue ratio of a game with repetition is

\[ P = \frac{\mathbb{E}[R]}{\mathbb{E}[T]} \]

Corollary

Let \( S_1 \) and \( S_2 \) be integrable non-stopping mining strategies. Strategy \( S_1 \) is more profitable than strategy \( S_2 \) if and only if \( P(S_1) \geq P(S_2) \).
Notations

- Two groups of miners with relative hashrates $0 < q < \frac{1}{2} < p < 1$, $p + q = 1$.
- The block validation times $T'$ and $T$ are exponentially distributed random variables with respective parameters $\alpha'$ and $\alpha$.
- The probabilities of success of each group are $P[T < T'] = p$, $P[T' < T] = q$.
- $N'(t)$ and $N(t)$ are Poisson processes with respective parameters $\alpha'$ and $\alpha$.
  - $P[N(t) = n] = \left(\frac{\alpha t}{n!}e^{-\alpha t}\right)^n$,
  - $P[N'(t) = n] = \left(\frac{\alpha' t}{n!}e^{-\alpha' t}\right)^n$.
Notations

- Two group of miners with relative hashrates

\[ 0 < q < \frac{1}{2} < p < 1, \quad p + q = 1 \]
Notations

- Two group of miners with relative hashrates
  \[ 0 < q < 1/2 < p < 1, \quad p + q = 1 \]
- The block validation times \( T' \) and \( T \) are exponentially distributed random variables with resp. parameters \( \alpha' \) and \( \alpha \).
Notations

- Two group of miners with relative hashrates
  \[ 0 < q < 1/2 < p < 1, \quad p + q = 1 \]
- The block validation times \( T' \) and \( T \) are exponentially distributed random variables with resp. parameters \( \alpha' \) and \( \alpha \).
- The probabilities of success of each group are
  \[
  \mathbb{P}[T < T'] = p, \quad \mathbb{P}[T' < T] = q.
  \]
Notations

- Two group of miners with relative hashrates
  \[0 < q < 1/2 < p < 1, \quad p + q = 1\]

- The block validation times \(T'\) and \(T\) are exponentially distributed random variables with resp. parameters \(\alpha'\) and \(\alpha\).

- The probabilities of success of each group are
  \[
  \mathbb{P}[T < T'] = p, \quad \mathbb{P}[T' < T] = q.
  \]

- \(N'(t)\) and \(N(t)\) numbers of validated blocks at time \(t\) are Poisson processes with resp. parameters \(\alpha'\) and \(\alpha\).
Notations

- Two group of miners with relative hashrates
  \[ 0 < q < \frac{1}{2} < p < 1, \quad p + q = 1 \]

- The block validation times \( T' \) and \( T \) are exponentially distributed random variables with resp. parameters \( \alpha' \) and \( \alpha \).

- The probabilities of success of each group are
  \[ \mathbb{P}[T < T'] = p, \quad \mathbb{P}[T' < T] = q. \]

- \( N'(t) \) and \( N(t) \) numbers of validated blocks at time \( t \) are Poisson processes with resp. parameters \( \alpha' \) and \( \alpha \).

  \[ \mathbb{P}[N(t) = n] = \frac{(\alpha t)^n}{n!} e^{-\alpha t}, \quad \mathbb{P}[N'(t) = n] = \frac{(\alpha' t)^n}{n!} e^{-\alpha't} \]
Simple example: Honest strategy

• Duration of the cycle for the honest strategy is the stopping time: $\tau_H = T' \wedge T$.
• We compute $E[\tau_H] = \tau_0 = \frac{1}{\alpha + \alpha'}$.
• Therefore, if $b > 0$ is the block reward, $E[R] = p_0 + q b = qb$.
• The revenue ratio of the honest strategy is $P(H) = qb \tau_0$. 

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Simple example: Honest strategy

- Duration of the cycle for the honest strategy is the stopping time:

\[ \tau_H = T' \land T \]
Simple example: Honest strategy

- Duration of the cycle for the honest strategy is the stopping time:
  \[ \tau_H = T' \land T \]

- We compute \[ \mathbb{E}[\tau_H] = \tau_0 = \frac{1}{\alpha + \alpha'} \].
**Simple example: Honest strategy**

- Duration of the cycle for the honest strategy is the stopping time:
  \[ \tau_H = T' \land T \]

- We compute \( \mathbb{E}[\tau_H] = \tau_0 = \frac{1}{\alpha + \alpha'} \).

- Therefore, if \( b > 0 \) is the block reward, \( \mathbb{E}[R] = p.0 + q.b = qb \)
Simple example: Honest strategy

- Duration of the cycle for the honest strategy is the stopping time:
  \[ \tau_H = T' \land T \]

- We compute \[ \mathbb{E}[\tau_H] = \tau_0 = \frac{1}{\alpha + \alpha'} \].

- Therefore, if \( b > 0 \) is the block reward, \( \mathbb{E}[R] = p.0 + q.b = qb \)

- The revenue ratio of the honest strategy is
  \[ P(H) = \frac{qb}{\tau_0} \]
Selfish mining

\[ R(c) = N(c)b \text{ and has a duration } T(c). \]

If the probability of each cycle is \( p(c) \) then
\[
E[R] = \sum_c p(c)R(c) \quad E[T] = \sum_c p(c)T(c).\]

The combinatorics is involved and the computation of each \( T(c) \) involves conditional probabilities and iterated integrals... too complex! We need new tools and ideas...

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Selfish mining

- The combinatorics of each cycle $c$ of attack gives a revenue $R(c) = N(c)b$ and has a duration $T(c)$.
Selfish mining

- The combinatorics of each cycle $c$ of attack gives a revenue $R(c) = N(c)b$ and has a duration $T(c)$.
- If the probability of each cycle is $p(c)$ then

$$
\mathbb{E}[R] = \sum_c p(c)R(c)
$$

$$
\mathbb{E}[T] = \sum_c p(c)T(c)
$$
Selfish mining

• The combinatorics of each cycle $c$ of attack gives a revenue $R(c) = N(c)b$ and has a duration $T(c)$.

• If the probability of each cycle is $p(c)$ then

$$E[R] = \sum_c p(c)R(c)$$

$$E[T] = \sum_c p(c)T(c)$$

• The combinatorics is involved and the computation of each $T(c)$ involves conditional probabilities and iterated integrals... too complex! We need new tools and ideas...
Martingale Technique

Definition (Martingale and Stopping Time)

A martingale is a stochastic process \(N(t)\) with an adapted decreasing filtration \((\Sigma_t)\) such that \(N(t)\) is \(\Sigma_t\)-measurable, and for \(t > s\)

\[E\left[N(t) | \Sigma_s\right] = N(s).\]

A stopping time \(\tau\) is a random variable taking values in \(\mathbb{R}^+\) only depending on \((N(t))\) for \(t \leq \tau\).

Theorem (Doob's Stopping Time Theorem)

Let \((N(t))\) be a martingale and \(\tau\) be a bounded stopping time. Then we have

\[E[N(\tau)] = N(0).\]
Martingale Technique

Definition (Martingale and Stopping Time)

A martingale is a stochastic process \((N(t))_{t \in \mathbb{R}^+}\) with an adapted decreasing filtration \((\Sigma_t)_{t \in \mathbb{R}^+}\) such that \(N(t)\) is \(\Sigma_t\)-measurable, and for \(t > s\)

\[\mathbb{E}[N(t)|\Sigma_s] = N(s) .\]
Martingale Technique

Definition (Martingale and Stopping Time)

A martingale is a stochastic process \((N(t))_{t \in \mathbb{R}_+}\) with an adapted decreasing filtration \((\Sigma_t)_{t \in \mathbb{R}_+}\) such that \(N(t)\) is \(\Sigma_t\)-measurable, and for \(t > s\)

\[
\mathbb{E}[N(t)|\Sigma_s] = N(s).
\]
Martingale Technique

Definition (Martingale and Stopping Time)

A martingale is a stochastic process \((N(t))_{t \in \mathbb{R}^+}\) with an adapted decreasing filtration \((\Sigma_t)_{t \in \mathbb{R}^+}\) such that \(N(t)\) is \(\Sigma_t\)-measurable, and for \(t > s\)

\[
\mathbb{E}[N(t) | \Sigma_s] = N(s).
\]

A stopping time \(\tau\) is a random variable taking values in \(\mathbb{R}^+\) only depending on \((N(t))_{t \leq \tau}\).
Martingale Technique

Definition (Martingale and Stopping Time)

A martingale is a stochastic process \((N(t))_{t \in \mathbb{R}^+}\) with an adapted decreasing filtration \((\Sigma_t)_{t \in \mathbb{R}^+}\) such that \(N(t)\) is \(\Sigma_t\)-measurable, and for \(t > s\)

\[
\mathbb{E}[N(t)|\Sigma_s] = N(s) .
\]

A stopping time \(\tau\) is a random variable taking values in \(\mathbb{R}^+\) only depending on \((N(t))_{t \leq \tau}\).

Theorem (Doob’s Stopping Time Theorem)

Let \((N(t))_{t \in \mathbb{R}^+}\) be a martingale and \(\tau\) be a bounded stopping time.
Martingale Technique

Definition (Martingale and Stopping Time)

A martingale is a stochastic process \((N(t))_{t \in \mathbb{R}_+}\) with an adapted decreasing filtration \((\Sigma_t)_{t \in \mathbb{R}_+}\) such that \(N(t)\) is \(\Sigma_t\)-measurable, and for \(t > s\)

\[
\mathbb{E}[N(t)|\Sigma_s] = N(s) .
\]

A stopping time \(\tau\) is a random variable taking values in \(\mathbb{R}_+\) only depending on \((N(t))_{t \leq \tau}\).

Theorem (Doob’s Stopping Time Theorem)

Let \((N(t))_{t \in \mathbb{R}_+}\) be a martingale and \(\tau\) be a bounded stopping time.

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
# Martingale Technique

## Definition (Martingale and Stopping Time)

A martingale is a stochastic process \((N(t))_{t \in \mathbb{R}_+}\) with an adapted decreasing filtration \((\Sigma_t)_{t \in \mathbb{R}_+}\) such that \(N(t)\) is \(\Sigma_t\)-measurable, and for \(t > s\)

\[
\mathbb{E}[N(t)|\Sigma_s] = N(s).
\]

A stopping time \(\tau\) is a random variable taking values in \(\mathbb{R}_+\) only depending on \((N(t))_{t \leq \tau}\).

## Theorem (Doob’s Stopping Time Theorem)

Let \((N(t))_{t \in \mathbb{R}_+}\) be a martingale and \(\tau\) be a bounded stopping time. Then we have \(\mathbb{E}[N(\tau)] = N(0)\).
Poisson Games

Theorem (Poisson Races)

Let \( \alpha \) and \( \alpha' \) be the parameters of two independent Poisson processes \( N \) and \( N' \), respectively, with \( \alpha' < \alpha \) and \( N(0) = N'(0) = 0 \).

Then, the stopping time \( \tau = \inf\{t > 0; N(t) = N'(t) + 1\} \) is finite a.s. and integrable.

Moreover, we have

\[
E[\tau] = \frac{1}{\alpha - \alpha'}, \\
E[N'(\tau)] = \frac{\alpha'}{\alpha - \alpha'}, \\
E[N(\tau)] = \frac{\alpha}{\alpha - \alpha'}.
\]
Poisson Games

Theorem (Poisson Races)

\( N \) and \( N' \) two independent Poisson processes with parameters \( \alpha' \) and \( \alpha \) with \( \alpha' < \alpha \) and \( N(0) = N'(0) = 0 \).
Poisson Games

Theorem (Poisson Races)

$N$ and $N'$ two independent Poisson processes with parameters $\alpha'$ and $\alpha$ with $\alpha' < \alpha$ and $N(0) = N'(0) = 0$. 

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Poisson Games

Theorem (Poisson Races)

\( N \) and \( N' \) two independent Poisson processes with parameters \( \alpha' \) and \( \alpha \) with \( \alpha' < \alpha \) and \( N(0) = N'(0) = 0 \). Then, the stopping time

\[
\tau = \inf\{t > 0; N(t) = N'(t) + 1\}
\]

is finite a.s. and integrable.
Theorem (Poisson Races)

Let \( \mathcal{N} \) and \( \mathcal{N}' \) be two independent Poisson processes with parameters \( \alpha' \) and \( \alpha \) with \( \alpha' < \alpha \) and \( \mathcal{N}(0) = \mathcal{N}'(0) = 0 \). Then, the stopping time

\[
\tau = \inf\{t > 0; \mathcal{N}(t) = \mathcal{N}'(t) + 1\}
\]

is finite a.s. and integrable. Moreover, we have

\[
\mathbb{E}[\tau] = \frac{1}{\alpha - \alpha'}, \quad \mathbb{E}[\mathcal{N}'(\tau)] = \frac{\alpha'}{\alpha - \alpha'}, \quad \mathbb{E}[\mathcal{N}(\tau)] = \frac{\alpha}{\alpha - \alpha'}.
\]
Proof

Proof.

Assume $\tau$ bounded
Proof

Proof.

Assume $\tau$ bounded
Proof

Proof. Assume $\tau$ bounded (otherwise truncate $\tau \land t_0$ and make $t_0 \to +\infty$).
Proof.

Assume $\tau$ bounded (otherwise truncate $\tau \land t_0$ and make $t_0 \to +\infty$). The compensated Poisson processes $N(t) - \alpha t$ and $N'(t) - \alpha' t$ are martingales.
Proof.

Assume $\tau$ bounded (otherwise truncate $\tau \land t_0$ and make $t_0 \to +\infty$). The compensated Poisson processes $N(t) - \alpha t$ and $N'(t) - \alpha' t$ are martingales. Doob’s Stopping Time Theorem gives
Proof

Assume $\tau$ bounded (otherwise truncate $\tau \wedge t_0$ and make $t_0 \to +\infty$). The compensated Poisson processes $N(t) - \alpha t$ and $N'(t) - \alpha' t$ are martingales. Doob’s Stopping Time Theorem gives

$$\alpha \mathbb{E}[\tau] = \mathbb{E}[N(\tau)] = \mathbb{E}[N'(\tau)] + 1 = \alpha' \mathbb{E}[\tau] + 1$$
Proof

Proof.

Assume $\tau$ bounded (otherwise truncate $\tau \land t_0$ and make $t_0 \to +\infty$). The compensated Poisson processes $N(t) - \alpha t$ and $N'(t) - \alpha' t$ are martingales. Doob's Stopping Time Theorem gives

$$\alpha \mathbb{E}[\tau] = \mathbb{E}[N(\tau)] = \mathbb{E}[N'(\tau)] + 1 = \alpha' \mathbb{E}[\tau] + 1$$

from where we get

$$\mathbb{E}[\tau] = \frac{1}{\alpha - \alpha'}$$

and the two other formulas.
Selfish Mining Stopping Time

Lemma (Duration of attack cycles)

The duration of attack cycles for selfish mining is given by the stopping time

\[ \tau_{SM} = \inf \{ t \geq T_1; N(t) = N'(t) - 1 + 2 \cdot 1 < T'_1 < T_1 < S_2 < S'_2 \} \]
Selfish Mining Stopping Time

- Let $T_1, T_2, \ldots$ and $T'_1, T'_2, \ldots$ interblock validation times.
Selfish Mining Stopping Time

- Let $T_1, T_2, \ldots$ and $T'_1, T'_2, \ldots$ interblock validation times.
- $S_n = T_1 + \ldots + T_n$, $S'_n = T'_1 + \ldots + T'_n$. 

**Lemma (Duration of attack cycles)**

The duration of attack cycles for selfish mining is given by the stopping time

$$\tau_{SM} = \inf \{ t \geq T_1; N(t) = N'(t) - 1 + 2 \cdot \frac{1}{T_1} < T'_1 + 2 \cdot \frac{1}{T'_1} < T_1 < S_2 < S'_2 \}$$
Selfish Mining Stopping Time

- Let \( T_1, T_2, \ldots \) and \( T'_1, T'_2, \ldots \) interblock validation times.
- \( S_n = T_1 + \ldots + T_n \), \( S'_n = T'_1 + \ldots + T'_n \).

**Lemma (Duration of attack cycles)**

The duration of attack cycles for selfish mining is given by the stopping time

\[
\tau_{SM} = \inf \{ t \geq T_1; N(t) = N'(t) - 1 + 2 \cdot 1_{T_1 < T'_1} + 2 \cdot 1_{T'_1 < T_1 < S_2 < S'_2} \}
\]
Selfish Mining Revenue Ratio

Theorem (SM Revenue Ratio)

\[ \tau_{SM} \text{ and } R(\tau_{SM}, \gamma) \text{ are integrable and} \]

\[ E[R(\tau_{SM})] = \left( 1 + pq \right) \left( p - q \right) + pq \frac{p}{p - q} \frac{qb}{\tau_0} - \left( 1 - \gamma \right) \frac{p^2}{\tau_0} \leq P(H) \]

\[ E[\tau_{SM}] = \left( 1 + pq \right) \left( p - q \right) + pq \frac{p}{p - q} \tau_0 - \left( 1 - \gamma \right) \frac{p^2}{q} \left( p - q \right) b \left( \left( 1 + pq \right) \left( p - q \right) + pq \right) \tau_0 \]
Selfish Mining Revenue Ratio

Theorem (SM Revenue Ratio)

$\tau_{SM}$ and $R(\tau_{SM}, \gamma)$ are integrable and
Selfish Mining Revenue Ratio

Theorem (SM Revenue Ratio)

\( \tau_{SM} \) and \( R(\tau_{SM}, \gamma) \) are integrable and

\[
E[\tau_{SM}] = (1 + pq)(p - q) + pq(p - q)\tau_0 \leq P(H)
\]

R. Pérez-Marco
On profitability of Selfish Mining (joint work with C. Grunspan)
Selfish Mining Revenue Ratio

Theorem (SM Revenue Ratio)

\(\tau_{SM}\) and \(R(\tau_{SM}, \gamma)\) are integrable and

\[
\mathbb{E}[R(\tau_{SM})] = \frac{(1 + pq)(p - q) + pq}{p - q} qb - (1 - \gamma)p^2 q b
\]

\[
\mathbb{E}[\tau_{SM}] = \frac{(1 + pq)(p - q) + pq}{p - q} \tau_0
\]
Selfish Mining Revenue Ratio

Theorem (SM Revenue Ratio)

\( \tau_{SM} \) and \( R(\tau_{SM}, \gamma) \) are integrable and

\[
\mathbb{E}[R(\tau_{SM})] = \frac{(1 + pq)(p - q) + pq}{p - q} q b - (1 - \gamma)p^2 q b
\]
\[
\mathbb{E}[\tau_{SM}] = \frac{(1 + pq)(p - q) + pq}{p - q} \tau_0
\]

and

\[
P(SM) = \frac{qb}{\tau_0} - (1 - \gamma)\frac{p^2 q(p - q)b}{((1 + pq)(p - q) + pq) \tau_0} \leq P(H)
\]
The Theorem from beyond

The following theorem shows that in a stable regime without difficulty adjustments the Bitcoin protocol is stable with respect to block withholding strategies.

Theorem (Optimality of Honest Mining)
For any block withholding strategy $S$ we have $P(S) \leq P(H) = q b \tau_0$.
The Theorem from beyond

The following theorem shows that in a stable regime without difficulty adjustments the Bitcoin protocol is stable with respect to block withholding strategies.
The following theorem shows that in a stable regime without difficulty adjustments the Bitcoin protocol is stable with respect to block withholding strategies.

**Theorem (Optimality of Honest Mining)**

*For any block withholding strategy S we have*

\[
P(S) \leq P(H) = \frac{qb}{\tau_0}
\]
Lead-Stubborn Mining Stopping Time

Lemma (Duration of attack cycles)
The duration of attack cycles for Lead-Stubborn mining is given by the stopping time 

\[ \xi_{LStM} = \tau + (T_N(\tau) + 1 \wedge T'_N(\tau) + 1) \cdot 1 \leq T_1 \]

with 

\[ \tau = \inf \left\{ t \geq T_1 : N(t) = N'(t) + 1 \wedge T_1 < T'_1 \right\} \]
Lemma (Duration of attack cycles)

The duration of attack cycles for Lead-Stubborn mining is given by the stopping time $\xi_{LStM}$

$$\xi_{LStM} = \tau + (T_{N(\tau)} + 1 \wedge T'_{N(\tau)} + 1) \cdot 1_{T'_1 \leq T_1}$$
Lead-Stubborn Mining Stopping Time

**Lemma (Duration of attack cycles)**

The duration of attack cycles for Lead-Stubborn mining is given by the stopping time $\xi_{LStM}$

$$
\xi_{LStM} = \tau + (T_{N(\tau)+1} \land T'_{N(\tau)+1}) \cdot 1_{T'_1 \leq T_1}
$$
Lemma (Duration of attack cycles)

The duration of attack cycles for Lead-Stubborn mining is given by the stopping time $\xi_{LStM}$

$$\xi_{LStM} = \tau + (T_{N(\tau)+1} \wedge T'_{N(\tau)+1}) \cdot 1_{T'_1 \leq T_1}$$

with

$$\tau = \inf\{t \geq T_1; N(t) = N'(t) + 1_{T_1 < T'_1}\}$$
Lead-Stubborn Mining Revenue Ratio

Theorem (LStM Revenue Ratio)

\[ \xi_{LStM} \text{ and } R(\xi_{LStM}) \]

are integrable and

\[
E[R(\xi_{LStM})] = \left( p + pq - q^2 \right) b - pq f b
\]

\[ E[\xi_{LStM}] = p + pq - q^2 \tau_0 \]

with

\[ f = 1 - \gamma \cdot \left( 1 - \frac{1}{2} q \cdot \left( 1 - \frac{1}{2} \sqrt{1 - 4(1 - \gamma) pq} \right) \right) \]

and

\[ P(\xi_{LStM}) = qb^\tau_0 - \left( p - q \right) \left( p + q \right) \tau_0 b \]

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Lead-Stubborn Mining Revenue Ratio

Theorem (LStM Revenue Ratio)

\( \xi_{LStM} \) and \( R(\xi_{LStM}) \) are integrable and...
Lead-Stubborn Mining Revenue Ratio

Theorem (LStM Revenue Ratio)

$\xi_{LStM}$ and $R(\xi_{LStM})$ are integrable and

\[
\mathbb{E}[\xi_{LStM}] = \frac{p + pq - q^2}{p - q^2 \tau_0} \quad \text{and} \quad P(\xi_{LStM}) = \frac{qb - pq f}{p + q(p - q) b \tau_0}
\]

R. Pérez-Marco
On profitability of Selfish Mining (joint work with C. Grunspan)
Lead-Stubborn Mining Revenue Ratio

**Theorem (LStM Revenue Ratio)**

\[ \xi_{LStM} \text{ and } R(\xi_{LStM}) \text{ are integrable and} \]

\[
\mathbb{E}[R(\xi_{LStM})] = \left( \frac{p + pq - q^2}{p - q} \right) qb - pq f b \]

\[
\mathbb{E}[\xi_{LStM}] = \frac{p + pq - q^2}{p - q} \tau_0
\]
Lead-Stubborn Mining Revenue Ratio

Theorem (LStM Revenue Ratio)

\( \xi_{LStM} \) and \( R(\xi_{LStM}) \) are integrable and

\[
\mathbb{E}[R(\xi_{LStM})] = \left( \frac{p + pq - q^2}{p - q} \right) qb - pq f b
\]

\[
\mathbb{E}[\xi_{LStM}] = \frac{p + pq - q^2}{p - q} \tau_0
\]

with \( f = \frac{1 - \gamma}{\gamma} \cdot \left( 1 - \frac{1}{2q} \left( 1 - \sqrt{1 - 4(1 - \gamma)pq} \right) \right) \)
Lead-Stubborn Mining Revenue Ratio

**Theorem (LStM Revenue Ratio)**

\( \xi_{LStM} \) and \( R(\xi_{LStM}) \) are integrable and

\[
\mathbb{E}[R(\xi_{LStM})] = \left( \frac{p + pq - q^2}{p - q} \right) qb - pq f b
\]

\[
\mathbb{E}[\xi_{LStM}] = \frac{p + pq - q^2}{p - q} \tau_0
\]

with \( f = \frac{1 - \gamma}{\gamma} \cdot \left( 1 - \frac{1}{2q} (1 - \sqrt{1 - 4(1 - \gamma)pq}) \right) \) and

\[
P(\xi_{LStM}) = \frac{qb}{\tau_0} - \frac{(p - q)pq f}{p + q(p - q)} \frac{b}{\tau_0}
\]

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Equal Fork Stubborn Stopping Time

Lemma (Equal Fork Stubborn Stopping Time)
The duration of attack cycles for Lead-Stubborn mining is given by the stopping time $\xi_{EFStM}$:

$$\xi_{EFStM} = \inf \{ t \geq 0 ; N(t) = N'(t) + 1 \}$$

Note: Same stopping time that for Poisson Games.

R. Pérez-Marco
On profitability of Selfish Mining (joint work with C. Grunspan)
Equal Fork Stubborn Stopping Time

**Lemma (Equal Fork Stubborn Stopping Time)**

*The duration of attack cycles for Lead-Stubborn mining is given by the stopping time* $\xi_{EFS\text{t}M}$

$$
\xi_{EFS\text{t}M} = \inf\{t \geq 0; N(t) = N'(t) + 1\}
$$
Equal Fork Stubborn Stopping Time

Lemma (Equal Fork Stubborn Stopping Time)

The duration of attack cycles for Lead-Stubborn mining is given by the stopping time $\xi_{EFStM}$

$$\xi_{EFStM} = \inf\{ t \geq 0; N(t) = N'(t) + 1 \}$$
Equal Fork Stubborn Stopping Time

Lemma (Equal Fork Stubborn Stopping Time)

The duration of attack cycles for Lead-Stubborn mining is given by the stopping time $\xi_{EFStM}$

$$\xi_{EFStM} = \inf\{ t \geq 0; N(t) = N'(t) + 1 \}$$

Note: Same stopping time that for Poisson Games.
Equal Fork Stubborn Revenue Ratio

\[ \xi_{\text{EFStM}} \quad \text{and} \quad R(\xi_{\text{EFStM}}) \]

are integrable and

\[ E[R(\xi_{\text{EFStM}})] = \frac{q}{p} - \frac{q}{b} - \frac{gb}{\tau_0} \]

with

\[ g = 1 - \gamma \gamma \left( 1 - \frac{1}{2} \left( 1 - \gamma \right) q \left( 1 - \sqrt{1 - 4 \left( 1 - \gamma \right) pq} \right) \right) \]

and

\[ P(\xi_{\text{EFStM}}) = \frac{qb}{\tau_0} - \frac{(p - q)gb}{\tau_0} \]

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Theorem (LStM Revenue Ratio)

\( \xi_{\text{EFS}tM} \) and \( R(\xi_{\text{EFS}tM}) \) are integrable and
Equal Fork Stubborn Revenue Ratio

Theorem (LStM Revenue Ratio)

\[ \xi_{EFS_{StM}} \text{ and } R(\xi_{EFS_{StM}}) \text{ are integrable and} \]

\[ E[ R(\xi_{EFS_{StM}}) ] = \frac{q}{p} - \frac{q}{p} b - \frac{gb}{\xi_{EFS_{StM}}} \]

\[ E[ \xi_{EFS_{StM}} ] = \frac{\tau_0}{p} - \frac{q}{p} g b \]

\[ \text{with} \quad g = 1 - \gamma \gamma (1 - \frac{1}{2} (1 - \gamma))^{\frac{q}{1 - \sqrt{1 - 4(1 - \gamma) p q}}}) \]

\[ P(\xi_{EFS_{StM}}) = \frac{qb}{\tau_0} - \frac{(p - q)gb}{\tau_0} \]
Equal Fork Stubborn Revenue Ratio

Theorem (LStM Revenue Ratio)

\( \xi_{EFS_{tM}} \) and \( R(\xi_{EFS_{tM}}) \) are integrable and

\[
\mathbb{E}[R(\xi_{EFS_{tM}})] = \frac{q}{p-q} b - gb \\
\mathbb{E}[\xi_{EFS_{tM}}] = \frac{\tau_0}{p-q}
\]
Equal Fork Stubborn Revenue Ratio

Theorem (LStM Revenue Ratio)

\( \xi_{\text{EFStM}} \) and \( R(\xi_{\text{EFStM}}) \) are integrable and

\[
\mathbb{E}[R(\xi_{\text{EFStM}})] = \frac{q}{p - q} b - gb
\]

\[
\mathbb{E}[\xi_{\text{EFStM}}] = \frac{\tau_0}{p - q}
\]

with \( g = \frac{1 - \gamma}{\gamma} \left( 1 - \frac{1}{2(1 - \gamma)q} \left( 1 - \sqrt{1 - 4(1 - \gamma)pq} \right) \right) \)

and

\[
P(\xi_{\text{EFStM}}) = \frac{qb}{\tau_0} - (p - q)g \frac{b}{\tau_0}
\]
Profitability after a difficulty adjustment
Profitability after a difficulty adjustment

- A block withholding attack that aims to orphan honest mined blocks slows down the network.
Profitability after a difficulty adjustment

- A block withholding attack that aims to orphan honest mined blocks slows down the network.
- After $n_0 = 2016$ (sic, 2015) blocks we have the difficulty divided by a factor $\delta > 1$. We need to re-evaluate the profitability.
Profitability after a difficulty adjustment

- A block withholding attack that aims to orphan honest mined blocks slows down the network.

- After \( n_0 = 2016 \) (sic, 2015) blocks we have the difficulty divided by a factor \( \delta > 1 \). We need to re-evaluate the profitability.

- For an attack cycle, \( \mathbb{E}[R] \) is unchanged but \( \mathbb{E}[T] \) is changed to \( \delta^{-1}\mathbb{E}[T] \) and the Revenue Ratio is multiplied by \( \delta \).
Profitability after a difficulty adjustment

- A block withholding attack that aims to orphan honest mined blocks slows down the network.

- After $n_0 = 2016$ (sic, 2015) blocks we have the difficulty divided by a factor $\delta > 1$. We need to re-evaluate the profitability.

- For an attack cycle, $\mathbb{E}[R]$ is unchanged but $\mathbb{E}[T]$ is changed to $\delta^{-1} \mathbb{E}[T]$ and the Revenue Ratio is multiplied by $\delta$.

- The apparent hashrate of the attackers becomes $\tilde{q} > q$. 
Profitability after a difficulty adjustment

- A block withholding attack that aims to orphan honest mined blocks slows down the network.

- After $n_0 = 2016$ (sic, 2015) blocks we have the difficulty divided by a factor $\delta > 1$. We need to re-evaluate the profitability.

- For an attack cycle, $\mathbb{E}[R]$ is unchanged but $\mathbb{E}[T]$ is changed to $\delta^{-1}\mathbb{E}[T]$ and the Revenue Ratio is multiplied by $\delta$.

- The apparent hashrate of the attackers becomes $\tilde{q} > q$.

- Depending on $q$ and $\gamma$ selfish mining strategies can become profitable.
BIP countermeasure against block withholding

The problem comes from the difficulty adjustment formula that ignores orphan blocks and therefore sub-estimates the total hashrate of the network.

It would be enough to include in honest validated blocks “proof of orphans”.

This could be done by propagating orphan headers through the network and including the data in validated blocks.

Then the new adjustment formula will take the ratio of the difference of first and last timestamp in a $n_0$ period and divide it by $n_0 + n'$ where $n'$ is the number of orphan blocks.
BIP countermeasure against block withholding

- The problem comes from the *difficulty adjustment formula* that ignores *orphan blocks* and therefore sub-estimates the total hashrate of the network.
BIP countermeasure against block withholding

- The problem comes from the difficulty adjustment formula that ignores orphan blocks and therefore sub-estimates the total hashrate of the network.

- It would be enough to include in honest validated blocks “proof of orphans”.

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
BIP countermeasure against block withholding

- The problem comes from the difficulty adjustment formula that ignores orphan blocks and therefore sub-estimates the total hashrate of the network.

- It would be enough to include in honest validated blocks “proof of orphans”.

- This could be done by propagating orphan headers through the network and include the data in validated blocks.
BIP countermeasure against block withholding

- The problem comes from the **difficulty adjustment formula** that ignores **orphan blocks** and therefore sub-estimates the total hashrate of the network.

- It would be enough to include in honest validated blocks “proof of orphans”.

- This could be done by propagating orphan headers through the network and include the data in validated blocks.

- Then the new adjustment formula will take the ratio of the difference of first and last timestamp in a $n_0$ period and divide it by $n_0 + n'$ where $n'$ is the number of orphan blocks.

R. Pérez-Marco

On profitability of Selfish Mining (joint work with C. Grunspan)
Apparent hashrates

Theorem (Apparent hashrates)

\[ \tilde{q} \left( SM \right) = \left( \left( 1 + pq \right) \left( p - q \right) + pq \right) q - \left( 1 - \gamma \right) p^2 q \left( p - q \right) + \left( p - q \right) \tilde{q} \left( LStM \right) = q \cdot p + pq - q^2 p + pq - q - pq \left( p - q \right) f \left( \gamma \right) \]

\[ \tilde{q} \left( EFStM \right) = q \cdot p - \left( 1 - \frac{q}{p} \right) f \left( \gamma \right) \]
Apparent hashrates

Theorem (Apparent hashrates)

\[
\tilde{q}(SM) = \frac{((1 + pq)(p - q) + pq)q - (1 - \gamma)p^2q(p - q)}{p^2q + p - q}
\]

\[
\tilde{q}(LStM) = q \cdot \frac{p + pq - q^2}{p + pq - q} - \frac{pq(p - q)f(\gamma)}{p + pq - q}
\]

\[
\tilde{q}(EFStM) = \frac{q}{p} - \left(1 - \frac{q}{p}\right)f(\gamma)
\]
Expected difficulty adjustments

Theorem (Expected difficulty adjustments)

\[ E[\delta (SM)] = p - q + pq(p - q) + pq^2 \]

\[ E[\delta (LStM)] = p + pq - q^2 \]

\[ E[\delta (EFStM)] = 1 \]
Expected difficulty adjustments

Theorem (Expected difficulty adjustments)

\[ E[\delta(SM)] = \frac{p - q + pq(p - q) + pq}{p^2q + p - q} \]

\[ E[\delta(LStM)] = \frac{p + pq - q^2}{p + pq - q} \]

\[ E[\delta(EFStM)] = \frac{1}{p} \]
Comparisons after a difficulty adjustment

For different values of $q$ and $\gamma$ we can compare the different strategies after a difficulty adjustment:
Comparisons after a difficulty adjustment

For different values of $q$ and $\gamma$ we can compare the different strategies after a difficulty adjustment:
Comparisons after a difficulty adjustment

For different values of $q$ and $\gamma$ we can compare the different strategies after a difficulty adjustment:
Comparisons with NKMS2016
Comparisons with NKMS2016
Thank you for your attention!