Análise do protocolo *Committeeless Proof-of-Stake*: em busca de um melhor ponto de operação

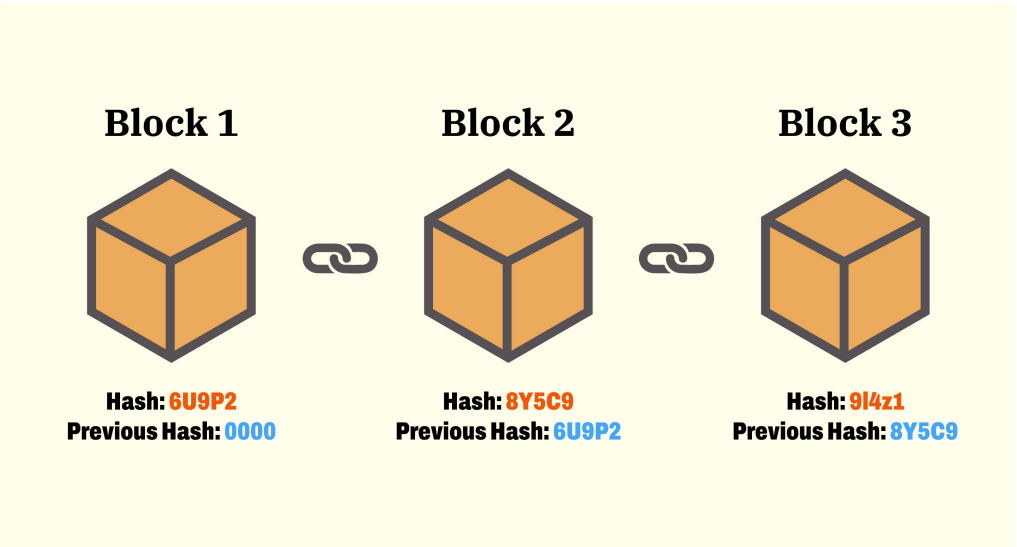
Vinícius Peixoto[1] Marco Aurélio Amaral Henriques[1]

[1]Faculdade de Engenharia Elétrica e de Computação (FEEC) - Unicamp

18 de Setembro de 2023

O que são blockchains?

- Livro-razão distribuído, imutável, aberto e descentralizado
- Unidade básica de dados: **bloco**
- Blocos são criados em intervalos definidos de tempo: **rodada**
- Blocos são ligados matematicamente entre si



O que são blockchains?

- Blockchains como bancos de dados distribuídos: redes peer-to-peer
- Grande quantidade de nós na rede, porém **apenas um bloco por rodada**
- Necessidade de sincronização
- **Mecanismo de consenso distribuído**

Mecanismos de consenso

Mecanismos mais utilizados:

- Proof-of-Work: Bitcoin, Litecoin, Monero, ...
- Proof-of-Stake: Ethereum, BNB, Cardano, ...

Proof-of-Work (PoW):

- Solução de um problema computacionalmente caro
- Quem resolver primeiro publica o bloco
- Grande desperdício de energia: trabalho de todos os nós exceto o sorteado é jogado fora

Mecanismos de consenso

Proof-of-Stake:

- Alternativa mais energeticamente eficiente ao PoW
- Nós investem *stake* para se tornarem **validadores**
- Validadores são sorteados para gerar blocos
- Validadores divididos em **comitês de votação**
- Comitês executam o protocolo de consenso e elegem blocos

Mecanismos de consenso

- Problema:
  - Centralização
  - Comitês de validação são superfícies de ataque
- Diversos ataques propostos: [Neuder 2021], [Schwarz-Schilling 2022]
  - Estima-se que organizações com menos de 1/3 do stake consigam comprometer o consenso
- Pergunta: é possível chegar ao consenso de forma **segura** e **descentralizada**?

O mecanismo CPoS

Committeeless Proof-of-Stake (CPoS):

- Eliminação da necessidade de um comitê
- Trabalho de criação, propagação e validação de nós é completamente distribuído
- Rede tenta convergir para um consenso de forma totalmente distribuída

O mecanismo CPoS

Este trabalho: investigação preliminar da influência de parâmetros de configuração na segurança do protocolo

Mecanismo de sorteio

Sorteio determinístico:

- Baseado no esquema Algorand [Gilad, 2017]
- Em um sorteio aleatório justo, seja w_i o número de fichas (*stake*) de um nó. Seja p a chance de uma dada ficha ser sorteada. Então a chance de exatamente k entre as w_i fichas serem sorteadas é dada pela distribuição binomial:

$$B(w_i, k, p) = \binom{w_i}{k} p^k (1-p)^{w_i - k}$$

- Sorteio: a partir de um conjunto de hashes, é calculado um número $q \in [0.0, 1.0]$. O total de fichas sorteadas é o maior valor k tal que $q > B(w_i, k, p)$.

Mecanismo de sorteio

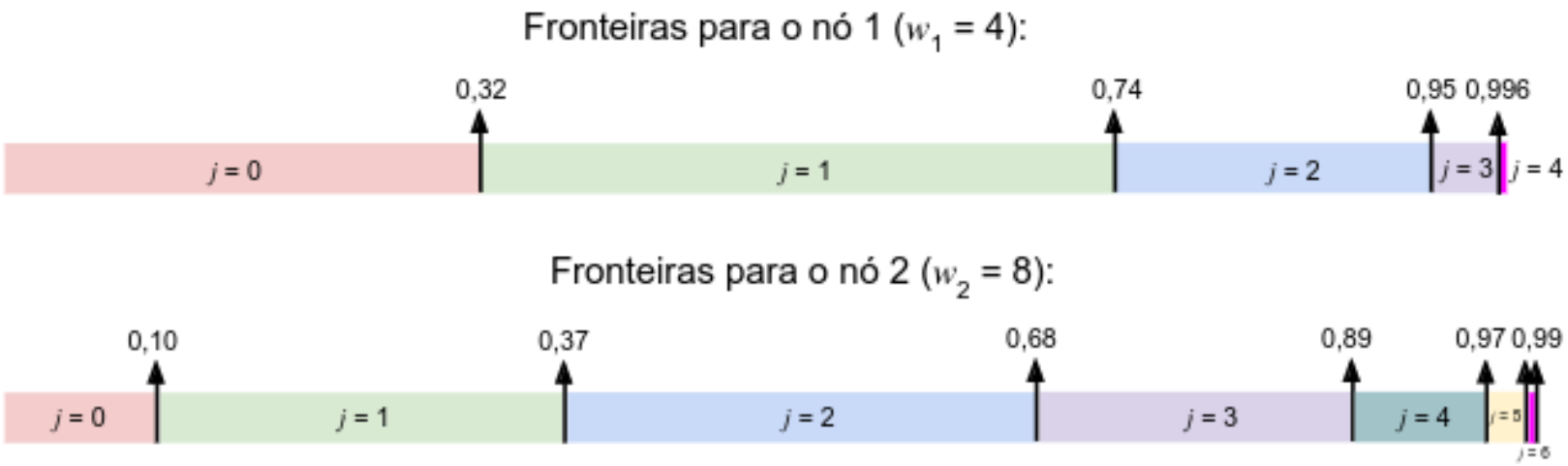


Figura 1: Subdivisão de intervalos para nós com stakes diferentes.

Mecanismo de sorteio

- Seja $W = \sum w_i$ o *stake* total na rede
- É possível provar que o número esperado **total de sorteios bem-sucedidos** por rodada é dado por $\tau = p \times W$

Mecanismo de sorteio

- Seja $W = \sum w_i$ o *stake* total na rede
- É possível provar que o número esperado **total de sorteios bem-sucedidos** por rodada é dado por $\tau = p \times W$
- Parâmetro τ: configura o número de blocos gerados por rodada

Confirmação de blocos

- Processo probabilístico

Confirmação de blocos

- Processo probabilístico
- Nós estimam um nível de confiança para cada bloco não-confirmado

Confirmação de blocos

- Processo probabilístico
- Nós estimam um nível de confiança para cada bloco não-confirmado
- Nível de confiança no bloco aumenta conforme chegam outros blocos que descendem dele

Confirmação de blocos

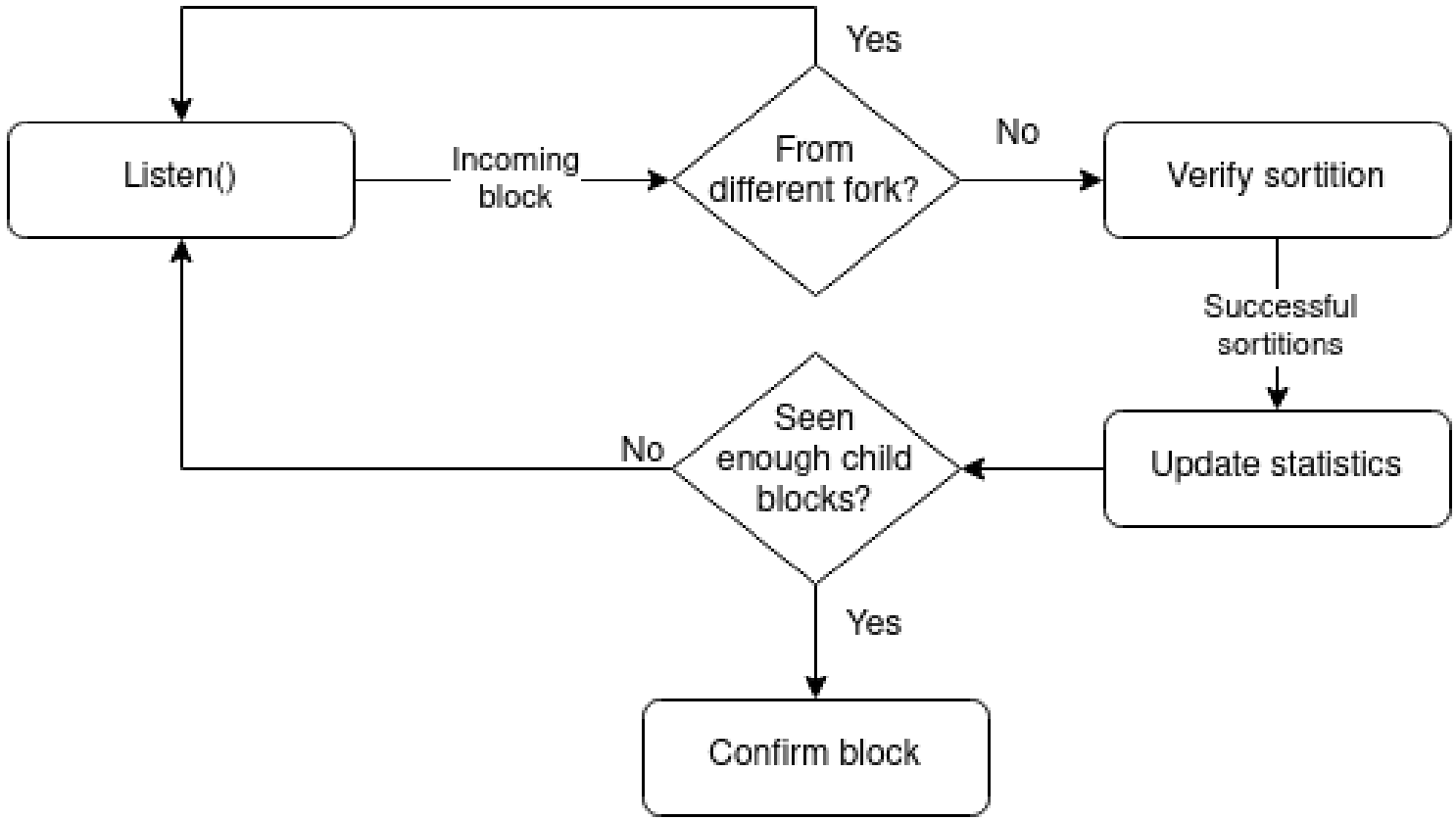


Figura 2: Fluxograma do algoritmo de confirmação de blocos.

Confirmação de blocos

Mecanismo de confirmação: exige recebimento de blocos

- Parâmetro τ controla o número total de blocos gerados

Confirmação de blocos

Mecanismo de confirmação: exige recebimento de blocos

- Parâmetro τ controla o número total de blocos gerados
- Possível ataque: nós desonestos não divulgam blocos

Confirmação de blocos

Mecanismo de confirmação: exige recebimento de blocos

- Parâmetro τ controla o número total de blocos gerados
- Possível ataque: nós desonestos não divulgam blocos
- **Investigação deste trabalho**: influência de τ na performance e segurança da rede

Experimentos

Experimentos

- Experimento 1: influência de τ em uma rede saudável
  - 25 peers no total
  - Cada peer conhece outros 5 peers aleatórios
  - Peers honestos (divulgam blocos)

Experimentos

- Experimento 1: influência de τ em uma rede saudável
  - 25 peers no total
  - Cada peer conhece outros 5 peers aleatórios
  - Peers honestos (divulgam blocos)
- Experimento 2: influência de τ em uma rede desonesta
  - 30 peers no total
  - 5 deles ($\approx 16\%$) não divulgam nós (desonestos)
  - Topologia de rede conexa

Experimentos

- 50 rodadas no total
- Média ao longo de 10 repetições de cada experimento
- Nós geram blocos vazios (somente headers)
- Infraestrutura de Docker, rodando no Linux 6.4, AMD Ryzen 7 3700X, 32GB RAM
- Código disponível em `https://github.com/regras/cpos_v2`

Resultados

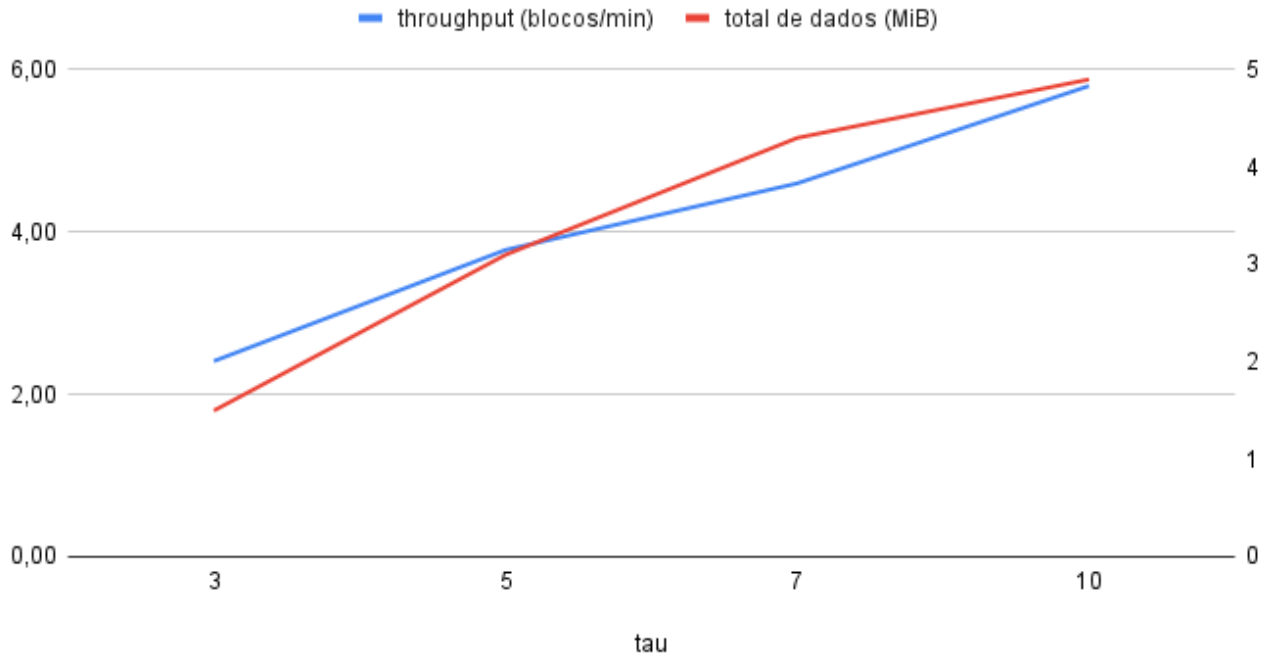


Figura 3: Influência de τ numa rede CPoS honesta.

Resultados

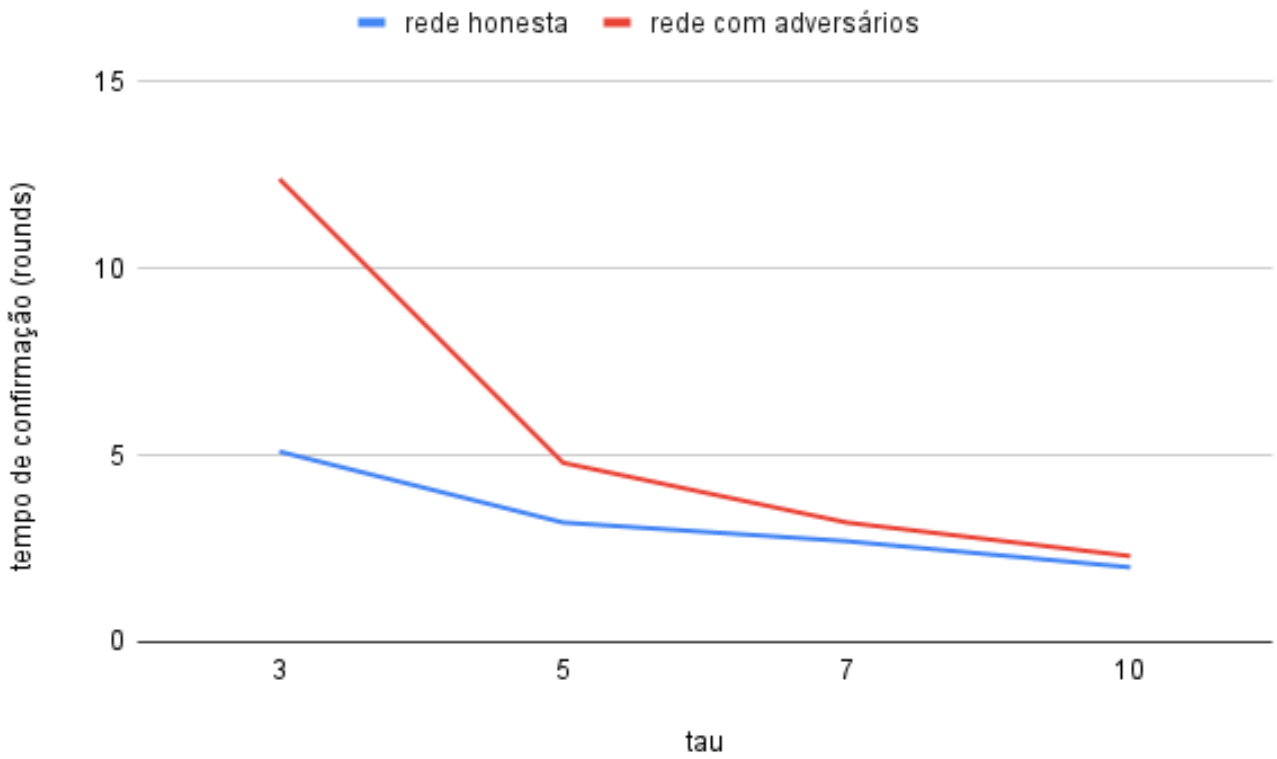


Figura 4: Influência de τ no tempo de confirmação do CPoS em uma rede saudável vs. uma rede adversarial

Conclusões

- Aumento de τ:

Conclusões

- Aumento de τ:
  - Maior throughput, menor tempo de confirmação
  - Aumento da resiliência em presença de nós desonestos

Conclusões

- Aumento de τ:
  - Maior throughput, menor tempo de confirmação
  - Aumento da resiliência em presença de nós desonestos
  - Contudo: aumento significativo no número de mensagens e total de dados em circulação

Conclusões

- Aumento de τ:
  - Maior throughput, menor tempo de confirmação
  - Aumento da resiliência em presença de nós desonestos
  - Contudo: aumento significativo no número de mensagens e total de dados em circulação
- Necessidade de encontrar um equilíbrio entre o valor de τ e o impacto na rede

Conclusões

- Aumento de τ:
  - Maior throughput, menor tempo de confirmação
  - Aumento da resiliência em presença de nós desonestos
  - Contudo: aumento significativo no número de mensagens e total de dados em circulação
- Necessidade de encontrar um equilíbrio entre o valor de τ e o impacto na rede
  - Envio somente de headers até que a rede escolha um bloco; somente então divulgação de blocos ocorre

Conclusões

Trabalhos futuros:

- Polimentos e melhorias na implementação atual do CPoS
- Execução de testes mais extensivos (maior número de blocos, nós distribuídos geograficamente)
- Busca de estratégias para minimização do consumo de dados do protocolo

Obrigado!

- Repositório do projeto:
  `https://github.com/regras/cpos_v2`
- E-mail para contato: `nukelet64@gmail.com`

Table 1. Relation between τ and blockchain performance/network stress.

τ	Blocks/min	Confirmation delay (rounds)	Total messages	Total data
3	2.41	5.1	2.8×10^3	1.5 MiB
5	3.78	3.2	5.8×10^3	3.1 MiB
7	4.61	2.7	8.0×10^3	4.3 MiB
10	5.87	2.0	9.1×10^3	5.1 MiB

Table 2. Relation between τ and confirmation delay on an adversarial network.

τ	Confirmation delay (rounds)
3	12.4
5	4.8
7	3.2
10	2.3

Confirmação de blocos:

- Baseada nos blocos que chegam até um nó
- O nó i calcula, na rodada x, o número total de sorteios bem-sucedidos nos blocos que recebeu: s_i^x
- Se os outros peers na rede estão no mesmo fork que i, ele espera ver em média τ sorteios bem sucedidos por rodada
- Nó calcula o número de sorteios médio: $\bar{s} = \frac{1}{\Delta_r} \sum s_i^x$
- Bloco confirmado quando $\bar{s}$ se torna suficientemente próximo de τ