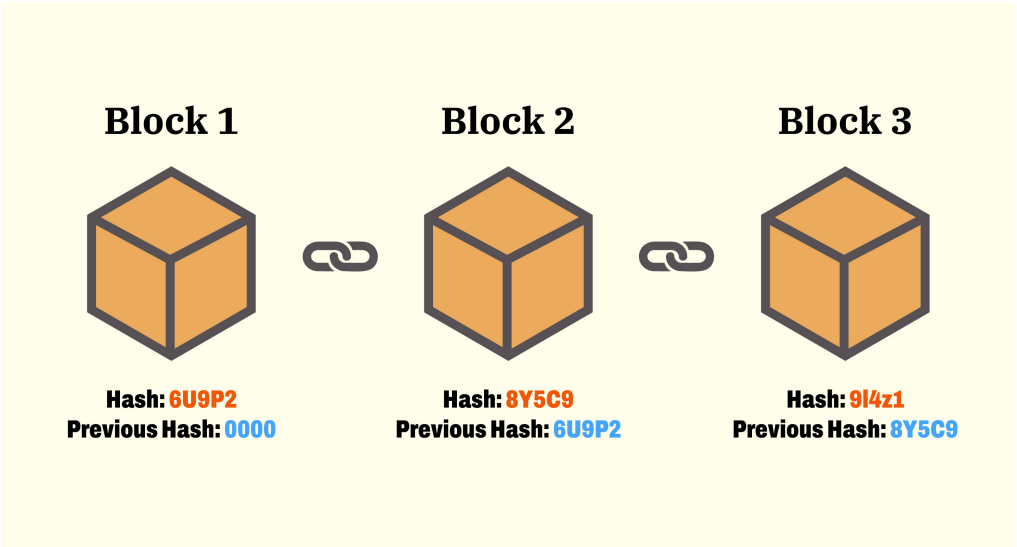
Analysis of the Committeeless Proof-of-Stake protocol: searching for a better point of operation

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

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

18 de Setembro de 2023

- 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



- 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**

Proof-of-Work (PoW):

- Solução de um problema computacionalmente caro
- Quem resolver primeiro publica o bloco
- Encontrar n tal que $h(n \parallel \text{block}) < 2^d$ (d ajustável)
- Grande desperdício de energia: trabalho de todos os nós exceto o sorteado é jogado fora

Proof-of-Stake:

- Alternativa mais energeticamente eficiente ao PoW
- Nós depositam uma quantidade de *stake* para fazerem parte de um *comitê de validação*
- Um membro do comitê é sorteado a cada rodada para gerar um bloco
- Membros fazem um processo de votação para validar e eleger o bloco

- Problema: o comitê de validação é uma superfície de ataque
- Foram propostos diversos ataques [Neuder 2021, Schwarz-Schilling 2022]
  - Estima-se que organizações com menos de 1/3 do stake consigam comprometer o consenso
- **Pergunta**: é possível chegar a um consenso sem um comitê de validação?

Committeeless Proof-of-Stake (CPoS):

- Ideia central: *verifiable random functions* $\rightarrow$ sorteios determinísticos
- Vários blocos gerados por rodada; critério determinístico de desambiguação
- Todos os nós na rede são responsáveis por validar e propagar blocos
- Rede tenta convergir para um consenso de forma totalmente distribuída

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 $q > B(w_i, k, p)$.

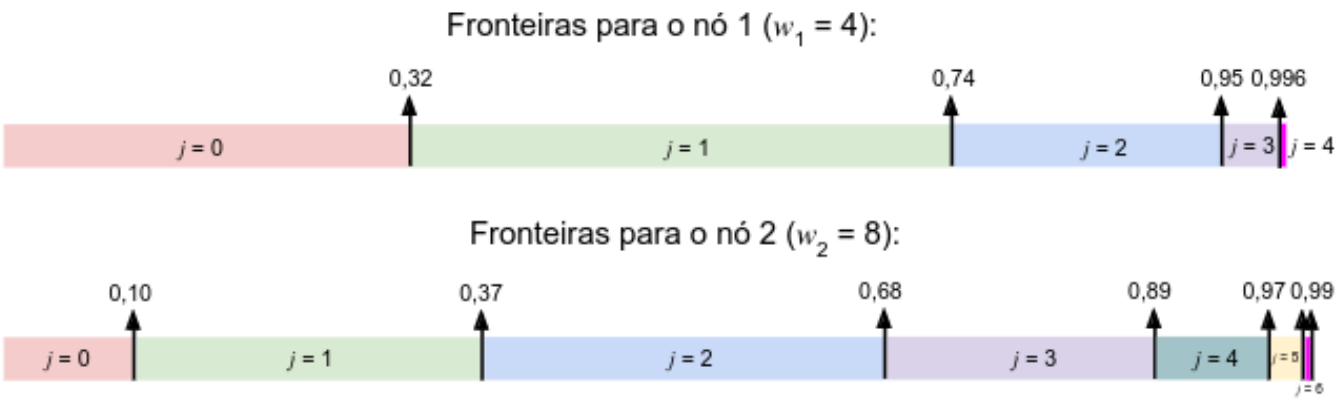


Figura 4.3 – Com o dobro do *stake*, o nó 2 tem maior probabilidade de ser sorteado, já que existem mais subintervalos e os valores das fronteiras são menores.

- Seja $W = \sum w_i$ o *stake* total na rede
- É possível provar que o número esperado total de blocos gerados por rodada é dado por $\tau = p \times W$
- Desse modo, p é calculado a partir do parâmetro de configuração τ

Confirmação de blocos:

- Baseada nos blocos que chegam até um nó
- O nó i calcula, na rodadada 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

- O processo de confirmação exige que os nós recebam uma quantidade suficientemente grande de blocos
- Contudo, o parâmetro τ controla o número total de blocos gerados
- Além disso, um possível ataque: nós não divulgam blocos quando são sorteados
- Investigação deste trabalho: **influência de τ na performance e resiliência da rede**

- Experimento 1: influência de τ em uma rede saudável
  - 25 peers no total
  - Cada peer conhece outros 5 peers
  - 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

- Tempo de rodada de 5s
- Nós geram blocos vazios (somente headers)
- Média de 10 execuções para cada experimento
- Infraestrutura de Docker, rodando no Linux 6.4, AMD Ryzen 7 3700X, 32GB RAM
- Código disponível em `https://github.com/regras/cpos_v2`

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

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

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
- Desenvolvimento de estratégias para punição de nós desonestos

Obrigado!