Análise comparativa da eficiência de mecanismos de consenso Proof-of-Stake com e sem comitês de validação

Matheus Acauã Dias , Marco Aurélio Amaral Henriques

[1]Faculdade de Engenharia Elétrica e de Computação (FEEC)
Universidade Estadual de Campinas (UNICAMP)
13083-852 – Campinas, SP, Brasil

m241617@dac.unicamp.br maah@unicamp.br

Abstract. *This work aims to analyze and compare the performance of two proof-of-stake consensus mechanisms for public blockchains: Casper (proposed by the Ethereum cryptocurrency group) and Committeeless Proof-of-Stake (CPoS, proposed by the ReGrAS/Unicamp group). The study is based on a theoretical and practical analysis of the two mechanisms, with the aim of identifying the advantages and limitations of each one, in order to offer a deeper understanding of these mecanisms by identifying constraints and enhancements.*

Resumo. *Este trabalho tem como objetivo analisar e comparar o desempenho de dois mecanismos de consenso Proof-of-Stake para blockchains públicas: Casper (proposto pelo grupo da criptomoeda Ethereum) e Committeeless Proof-of-Stake (CPoS, proposto pelo grupo ReGrAS/Unicamp). O estudo se baseia em uma análise teórica e prática dos dois mecanismos, com o intuito de identificar vantagens e limitações de cada um, a fim de oferecer uma compreensão mais aprofundada sobre esses mecanismos identificando limitações e melhorias.*

1. Introdução

A *blockchain* é um livro-razão distribuído que registra transações em uma rede. Diferentemente dos sistemas tradicionais que dependem de intermediários, a *blockchain* utiliza mecanismos de consenso para validar e manter registros. Esses mecanismos visam garantir a confiabilidade, segurança e integridade dos dados, permitindo que todas as partes concordem sobre o estado e a validade das transações [Saleh 2021].

Neste trabalho, fornecemos uma explicação concisa sobre os mecanismos de consenso, com foco nos algoritmos *Proof-of-Stake* (PoS). Partindo de uma visão geral do PoW e, em seguida, nos aprofundamos no PoS, abordando duas variações específicas: o Casper, que utiliza comitês de validação na rede *Ethereum*, e o CPoS, que foi proposto por nosso grupo de pesquisa e não depende de comitês. Após essa explanação, analisamos o desempenho de cada mecanismo, levando em consideração a vazão de cada um. O objetivo é identificar as limitações de cada mecanismo e possíveis melhorias.

2. Mecanismos de Consenso

Um mecanismo de consenso em uma rede *blockchain*, é um conjunto de regras que permite que os participantes concordem sobre o estado e a validade das transações [Bashir 2017]. Ele desempenha um papel fundamental na segurança, confiabilidade e

integridade da *blockchain*, prevenindo fraudes. Existem diferentes mecanismos de consenso, como o *Proof-of-Work* (PoW) e o *Proof-of-Stake* (PoS), que são utilizados para alcançar esse acordo. A eficiência de um consenso é medida pela sua capacidade de confirmar um grande número de transações por segundo e manter a segurança da *blockchain*.

2.1. Proof-of-Stake

Proof-of-Stake (PoS) é uma alternativa ao PoW, com validação de blocos e transações baseadas na participação (financeira) dos nós na rede, em vez do poder computacional. Os nós que colocam mais *stake* (apostam mais alto) na rede têm mais chances de serem selecionados para criar blocos e receber recompensas, incentivando a participação e aumentando a segurança do sistema. É como se o *stake* fosse o número de bilhetes de loteria com os quais um participante está concorrendo a um sorteio. No PoS só se faz um sorteio a cada certo intervalo de tempo (rodada) e, por isso, ele é mais eficiente em termos de consumo de energia, não exigindo computação intensiva e nem hardware especial. A maioria dos mecanismos PoS garante a lisura do processo de validação de blocos por meio de comitês de validação, os quais necessitam gerenciamento. Para evitar problemas de corrupção e/ou conluio entre os membros do comitê, alguns protocolos PoS implementam a seleção aleatória de membros, rotação de validadores no comitê ou substituição de comitês de validação por métodos probabilísticos, como é o caso do CPoS.

2.1.1. Particularidades do Casper como mecanismo PoS

O Casper é um mecanismo de consenso proposto para a rede *Ethereum*, uma das mais conhecidas *blockchains* do mundo, nele os blocos são produzidos por um conjunto de validadores selecionados de maneira aleatória, segundo um esquema de seleção proporcional ao *stake* depositado. Os *checkpoints* são blocos que definem uma *epoch* (ou época) e que estão posicionados em múltiplos predefinidos dentro da *blockchain*, como cada 32 blocos [epoch 2023]. A finalização desses *checkpoints* é realizada pela obtenção de $\frac{2}{3}$ dos votos emitidos pelos validadores, garantindo a confirmação dos blocos anteriores.

Casper define *"supermajority links"*: ocorrem quando mais de $\frac{2}{3}$ dos validadores votam em um link que vai do *checkpoint* s ao *checkpoint* t, validando todos os blocos nesse intervalo. Um *checkpoint* t é considerado justificado se ele for o bloco gênesis ou se o *checkpoint* s for justificado e o link $s \rightarrow t$ for um *"supermajority link"*. Um *checkpoint* t é finalizado se for justificado e existir um *"supermajority link"* $t \rightarrow u$, onde u é o *checkpoint* sucessor direto de t, isto é, está 32 blocos adiante. O fato de Casper se basear em comitês de validação traz alguns problemas de gerenciamento de tais comitês, já que na prática eles são dinâmicos, permitindo que alguns tipos de ataques sejam possíveis, como revisões de longo prazo e falhas catastróficas [Buterin and Griffith 2017].

2.1.2. Particularidades do CPoS como mecanismo PoS

A proposta do CPoS é eliminar a necessidade de um comitê de validação por meio de um consenso probabilístico [Martins 2021]. Nesse mecanismo, os nós são sorteados de forma aleatória e proporcional ao seu *stake* para produzir os blocos. Quando um nó recebe um bloco, ele verifica alguns critérios para determinar sua aceitação ou rejeição. Primeiro,

verifica se o bloco foi criado dentro do intervalo de tolerância em relação à rodada atual e, em seguida, se a rodada registrada no bloco recebido é menor que a rodada de um bloco recebido anteriormente (se houver tal bloco). Se esses critérios não forem suficientes para decidir qual bloco deve ser considerado, o bloco aceito será aquele com o menor *hash* de prova, um *hash* baseado em vários parâmetros que serve como critério de desempate entre blocos similares. Esse processo tem como objetivo evitar a criação e o prolongamento de *forks* na *blockchain* por mais de uma rodada.

O protocolo garante uma confirmação precisa em um cenário ideal onde todos os blocos produzidos são recebidos por toda a rede dentro dos limites definidos pelo intervalo de tolerância. No entanto, em situações do mundo real, com atrasos e falhas de transmissão na rede e nós desonestos, diferentes nós podem ter visões divergentes da *blockchain*. Portanto, o mecanismo realiza uma confirmação probabilística, em que um nó mede se sua visão atual da *blockchain* está sendo seguida pelos outros nós, sem a necessidade de um esquema de votação. O nó confirma um bloco quando a probabilidade de existirem visões conflitantes relacionadas a esse bloco é suficientemente baixa. O cálculo detalhado dessa probabilidade e o nível mínimo necessário para que um bloco seja confirmado são explicados na referência [Martins 2021].

3. Avaliação Comparativa entre Casper e CPoS

3.1. Desempenho Casper

O Casper é um mecanismo de consenso baseado em participação, onde a rapidez na finalização dos *checkpoints* e o tempo médio entre os blocos influenciam seu desempenho [Buterin and Griffith 2017]. No melhor cenário, os nós compartilham a mesma visão da *blockchain* e o Casper utiliza dois períodos para justificar e finalizar um *checkpoint*. O tempo médio entre os blocos multiplicado pelo número de blocos em um período determina o tempo necessário para justificar e finalizar o próximo *checkpoint*.

A Tabela 1 contém dados atuais do desempenho da rede *Ethereum*, os quais foram levantados para se fazer uma estimativa do desempenho do CPoS em uma situação semelhante. De acordo com essa tabela, retirada do site *blockchair.com*, podemos observar dados médios da rede *Ethereum* dos últimos 3 meses [Blockchair 2023]. Os valores em questão são a vazão, em transações por segundo (Tx/s), o tamanho médio do bloco, em kilobytes ($block_{size}$) e o número médio de transações por bloco. A última linha da Tabela 1 representa a média aritmética de cada coluna e essa média foi utilizada na seção seguinte para os cálculos e estimativas do CPoS. Com base nesses dados médios, podemos obter o tamanho médio das transações: $Tx_{size} = 113.406/149,6 = 758,1$ B. Deve ser notado que o Casper tem um tempo médio entre blocos (chamado de *block time*) de 12 segundos.

Tabela 1. Dados medidos da rede ethereum (Fonte: blockchair.com)

Data	Tx/s	$block_{size}(B)$	$Tx/block$
01/05/2023	13	102.487	155,0
01/06/2023	12	113.935	150,9
01/07/2023	12	123.797	142,9
Média	12,3	113.406	149,6

Outro parâmetro importante para o desempenho é o tempo de confirmação T_{conf}

das transações em um bloco, que depende do tempo necessário para que esse bloco seja definitivamente aceito pela *blockchain*, isto é, finalizado. Não está sendo considerado o tempo que os nós validadores podem levar para chegar a um consenso em suas votações, apenas o tempo médio entre blocos.

Uma transação é confirmada em um tempo mínimo se várias condições favoráveis são satisfeitas. A primeira é que seja inserida em um bloco s que é um *checkpoint*. A segunda é que esse *checkpoint* s seja votado pelo comitê (nem todos são) e seja justificado. Finalmente, é preciso que o próximo *checkpoint* t (exatamente 32 blocos adiante) também seja justificado, o que torna o bloco s finalizado (confirmado) [Buterin and Griffith 2017]. Assim, o tempo de confirmação mínimo $T_{conf,0}$ do bloco s e suas transações será o tempo de criação dos 32 blocos até t, o que resulta em $T_{conf,0} = 32 \cdot 12 = 384$s.

Em contraste, um segundo cenário ocorre quando uma transação é adicionada em um bloco x criado após um *checkpoint* s que já foi justificado. Nesse caso, o bloco x terá que esperar a criação de dois *"supermajority links"*: um entre s e um futuro *checkpoint* t, justificando t, e outro entre t e u, que precisa ser o sucessor direto de t (exatos 32 blocos adiante), finalizando t e confirmando todos os blocos e transações entre s e t (inclusive x). Esse bloco x pode estar em uma faixa entre $s + 1$ e $s + 31$. Dessa forma, o tempo de espera dependeria da posição de x em relação a s: se $x = s + 1$ (ou $x = s + 31$), seria necessário aguardar a criação de 31 (ou 1) blocos até t e mais outros 32 blocos até u, resultando em $T_{conf,1} = 63 \cdot 12 = 756$ (ou $T_{conf,1} = 33 \cdot 12 = 396$)s. Para fins de comparação, podemos tomar o caso médio, em que a transação é inserida em um bloco x equidistante de s e t. Nesse caso, $T_{conf,médio} = 48 \cdot 12 = 576$s.

3.2. Desempenho Committeeless Proof-of-Stake

A vazão do CPoS é dada por $Tx/s = \frac{block_{size}}{(Tx_{size} \times \Delta t)} \times Bc/round$ transações por segundo, onde $block_{size}$ e Tx_{size} têm o mesmo significado já definido, Δt é o período da rodada e $Bc/round$ é o número de blocos confirmados por rodada [Martins 2021]. O valor de $Bc/round$ depende de dois parâmetros básicos no CPoS: τ e ϵ.

O parâmetro τ é definido como $\tau = p \times W$, isto é, o produto da probabilidade p de um sorteio ser bem-sucedido pelo número total W de sorteios dentro de uma rodada e representa o número esperado de sorteios bem-sucedidos dentro de uma rodada. Um sorteio é bem-sucedido se ele produz um valor aceitável dentro dos parâmetros definidos pelo CPoS [Martins 2021].

O parâmetro ϵ é chamado de limiar de segurança e é estabelecido como o limite da probabilidade de haver mais sorteios bem-sucedidos em uma rodada que o valor τ. Ao estabelecer esse limite, temos uma visão clara sobre a probabilidade de blocos divergentes serem aceitos e criarem *forks* na rede. Ao ajustar esse limite de segurança, podemos aumentar ou diminuir o desempenho da nossa rede. Dessa forma, podemos adaptar a tolerância a *forks* indesejados, garantindo a estabilidade e a segurança do sistema.

Esses parâmetros atuam diretamente sobre o valor do $Bc/round$ e foram variados em diferentes execuções do CPoS com o fim de obter melhores taxas de transações por segundo [Martins 2021]. Na Tabela 2 são apresentados alguns resultados dessas execuções do CPoS em um ambiente formado por um conjunto de 25 hosts, distribuídos geograficamente. Cada host acomodou 16 nós da rede e o período de cada rodada foi $\Delta t = 20s$.

O tamanho do bloco neste caso particular foi de 1MB e o tamanho das transações foi de 400 B, o que resultou em uma média de 2500 transações por bloco (esses valores estão consistentes com os números obtidos em blockchains reais, como a do Bitcoin, por exemplo, e foram utilizados em diversos testes em [Martins 2021]). No CPoS, os parâmetros ϵ, τ e Δt determinam o tempo de confirmação de um bloco (T_{conf}), o qual é expresso em segundos e apresentado na última coluna da Tabela 2.

Tabela 2. Diferentes vazões e taxas de confirmação para combinações de τ e ϵ com base em dados de [Martins 2021]

τ	ϵ	Δt	Tx/s	$Bc/round$	T_{conf}
25	10^{-4}	20	90,00	0,72	27,78
25	10^{-3}	20	99,58	0,80	25,00
25	10^{-2}	20	114,43	0,92	21,74
19	10^{-2}	20	113,11	0,90	22,22
16	10^{-2}	20	97,56	0,78	25,64

3.3. Resultados e Análise

Para melhor entender as relações de eficiência em questão, focaremos, neste momento, na escalabilidade do mecanismo por meio do parâmetro de vazão. Os outros dois parâmetros importantes, segurança e descentralização, serão abordados em trabalhos futuros, após a conclusão da refatoração do CPoS, que está atualmente em andamento [ReGrAS 2023].

Com os parâmetros usados na Tab. 2, escolhidos de forma a otimizar o funcionamento do CPoS, nota-se que a vazão do mesmo varia entre 90 e 114,43 transações por segundo. Nesse cenário, a taxa de blocos confirmados por rodada varia de 0,72 a 0,92, não atingindo o máximo teórico de 1, devido ao fato de que em algumas rodadas podem não ocorrer sorteios bem-sucedidos. No caso em que se atinja esse patamar máximo, o número de transações por segundo alcança a marca de 125. Tais valores de vazão são cerca de 10 vezes superiores aos do Casper e isso se deve ao fato de se ter considerado um bloco de 1MB para o CPoS, que é um valor 8,8 vezes maior que o bloco do Casper. Essa diferença praticamente já justifica a diferença na vazão, ficando uma parte a cargo da diferença entre tempos de rodada, 20s. no CPoS contra cerca de 12s. no Casper, e outra parte a cargo da diferença de tamanho médio de transações (400 vs. 758,1 B).

Podemos fazer um exercício de transposição do CPoS para o cenário do Casper, adotando rodada de 12s e os tamanhos médios de bloco e de transações obtidos da Tab.1. Consideremos inicialmente o caso ideal de um bloco confirmado por rodada ($Bc/round = 1$): $Tx/s = \frac{113.406}{758{,}1 \times 12} \times 1 = 12,5$. Nesse caso, temos exatamente um décimo da vazão original do CPoS, o qual é um pouco superior à vazão média do Casper (12,3). Entretanto, se aplicarmos os valores reais de $Bc/round$ variando entre 0,72 e 0,92, temos que a vazão do CPoS em condições similares ao Casper varia na faixa de 9,0 a 11,5, valores um pouco abaixo da média deste último.

Uma possibilidade para alcançar um desempenho semelhante ou superior ao do Casper seria aumentar o tamanho do bloco e/ou diminuir o período de rodada Δt. Deve ser observado que o tamanho médio das transações depende dos clientes e está fora do controle dos nós que mantêm a *blockchain*. Deve ser adotado o pior caso (tamanho máximo

de transações permitido). A redução da rodada tem seus limites, como mostrado na referência [Martins 2021], pois a confirmação probabilística depende que grande parte dos nós recebam todos os blocos produzidos e, para tanto, é necessário que o valor de Δt seja suficiente para que os blocos trafeguem pela rede.

Comparando os tempos de confirmação de um bloco, observamos que o protocolo Casper possui uma latência média de 576s. Uma opção para reduzir esse tempo de latência, sem modificar o protocolo, seria diminuir o parâmetro $blocktime$, por exemplo, o que não é recomendável, pois isso torna o protocolo menos estável e seguro. Por outro lado, o protocolo CPoS, conforme apresentado na Tabela 2, possui uma latência entre 21,74 e 27,78s, sendo possível ajustá-la por meio de vários parâmetros, como ϵ, τ e Δt. Nota-se, portanto, que além do tempo de confirmação de um bloco do CPoS poder ser bem mais curto, este também pode ser ajustado por meio da regulagem dos parâmetros internos em busca de um ponto de operação mais otimizado.

4. Conclusões e trabalhos futuros

Com base nos dados de desempenho da rede *Ethereum*, foi observado que o Casper apresenta uma vazão média de transações por segundo cerca de dez vezes inferior ao CPoS, quando este último está trabalhando com parâmetros otimizados. No entanto, ao se adotar no CPoS parâmetros similares aos do Casper, vemos que a vazão do primeiro fica ligeiramente inferior á do segundo. Quanto ao tempo de confirmação de um bloco observamos que o CPoS possuí uma latência de confirmação bem menor e mais flexível que a do Casper. Foram discutidas maneiras de se melhorar o desempenho do CPoS, mas as mesmas precisam ser confirmadas em novos experimentos feitos em condições mais reais de operação da rede. A refatoração do código do CPoS [ReGrAS 2023], que está em andamento para corrigir alguns problemas operacionais e torná-lo mais eficiente permitirá a realização de testes e comparações mais sofisticados e próximos do contexto que temos na prática hoje para o Casper, principal mecanimo PoS em uso. Além disso, para uma análise mais completa, é necessário considerar outros aspectos, como segurança e descentralização e escalabilidade. Testes em diferentes cenários e casos de uso deverão ser realizados em trabalhos futuros para obter uma visão mais completa sobre a eficiência do CPoS em comparação, não só com o Casper, como também com outros mecanismos que utilizam de comites de validação.

Referências

Bashir, I. (2017). *Mastering Blockchain*. Packt Publishing Ltd., 1 ed edition.

Blockchair (2023). https://blockchair.com/pt/ethereum/charts.

Buterin, V. and Griffith, V. (2017). Casper the friendly finality gadget. ArXiv e-prints.

epoch, E. (2023). Url https://ethereum.org/en/glossary/#epoch .

Martins, D. F. G. (2021). Um novo mecanismo de consenso probabilístico para blockchains públicas. Disponível em: https://repositorio.unicamp.br/Busca/Download?codigoArquivo=507683 .

ReGrAS (2023). Repositório https://github.com/regras/cpos_v2 .

Saleh, F. (2021). Blockchain without waste: Proof-of-stake. Available at SSRN: https://ssrn.com/abstract=3183935 or http://dx.doi.org/10.2139/ssrn.3183935 .