Trabalho I de Computação Concorrente: Simulação de Mineração de Bitcoin

Gabriel da Fonseca Ottoboni Pinho - DRE 119043838
Rodrigo Delpreti de Siqueira - DRE 119022353

25/04/2021

1 Descrição do trabalho

1.1 Sobre a Bitcoin

Bitcoin é uma criptomoeda criada por Satoshi Nakamoto em 2009, tendo como objetivo ser uma moeda digital descentralizada. Uma parte essencial da Bitcoin é uma tecnologia chamada *Blockchain*, que consiste em armazenar todas as transações que já ocorreram em blocos interligados. Cada transação é criptograficamente assinada por seu emissor e então enviada ao resto da rede, que verifica a assinatura e a existência dos fundos.

Um ponto importante é que cada transação é verificada por todos, de modo que nenhum integrante precisa confiar em nenhum outro, sendo capaz de verificar a verdade independentemente. Os integrantes da rede que coletam as transações e as colocam em blocos são os mineradores e o sistema de consenso que determina qual minerador terá o direito de criar o bloco se chama *Proof of Work* (PoW).

A ideia da PoW é que o minerador que conseguir a resposta de um desafio primeiro tem o direito de criar o bloco. Esse desafio consiste em achar um número *nonce*, tal que quando o *hash* SHA-256 do *header* do bloco (que contém a *nonce*) seja menor que um certo número. Pelo fato do SHA-256 ser uma função *hash* criptográfica, a única forma de achar a *nonce* correta é chutando. Em outras palavras, não é possível achar um x tal que $\text{SHA256}(x) = y$. Por outro lado, tendo um x, é fácil verificar se $\text{SHA256}(x) = y$. Essa propriedade é importante, pois, dessa forma, todos os integrantes da rede podem verificar facilmente se a *nonce* encontrada é uma solução válida de fato.

1.2 Sobre o SHA-256

O SHA-256 é uma função *hash* criptográfica, que gera uma sequência de 256 bits pseudo-aleatória para uma entrada qualquer. O cálculo dessa função é computacionalmente custoso, e é *work* na PoW da Bitcoin. O cálculo da função consiste em 3 passos:

1. Pré-processamento
2. Criação do array de mensagens
3. Compressão

Durante o pré-processamento, a entrada é será divida em pedaços chamado *chunks* de 512 bits cada. Depois disso, para cada *chunk*, um array de 64 elementos de 4 bytes é preenchido com base no conteúdo do *chunk* atual. Por fim, 8 variáveis são inicializadas com valores pré-determinados que serão modificados em cada *round* do loop de compressão. O valor final do *hash* será a concatenação das 8 variáveis.

2 A solução

O objetivo do nosso trabalho é implementar o SHA-256 e usá-lo para simular a mineração da Bitcoin. Em outras palavras, vamos gerar 76 bytes aleatórios, que simularão o *header* de um bloco da Bitcoin. Em seguida, acrescentaremos mais 4 bytes, que representarão a *nonce*, totalizando 80 bytes. Depois, calcularemos o *hash* SHA-256 desses 80 bytes: se o resultado for menor que um certo número n, a *nonce* é uma solução válida. Senão, alteramos a *nonce* e tentamos novamente

É importante notar que cada tentativa é completamente independente da outra, ou seja, podemos facilmente acrescentar mais threads, com cada um testando diferentes valores para a *nonce* até que a resposta certa seja encontrada.