Three Dice Decentralized Consensus Algorithm

This consensus algorithm uses a probabilistic approach based on three dice rolls to achieve agreement in a public blockchain without a central authority. Here is a concise, four-step process for implementing this approach, along with probability calculations for different targets.

Step 1: Defining the Consensus Target

The consensus target is a numeric value that the participants must match or exceed through randomized attempts (dice rolls) to add a new block to the blockchain. Each participant independently rolls three dice to generate a random number between 3 and 18 (the range of sums for three six-sided dice). If their result matches or exceeds the target, they achieve consensus for that block and are rewarded.

- **Simple Target (e.g., 12)**: Participants attempt to achieve a target of 12 or higher by rolling three dice.
- **Difficult Target (e.g., 5)**: Participants attempt to achieve a target of 5 or lower by rolling three dice.

Step 2: Probability Calculations

To understand the likelihood of meeting specific targets, we calculate the probabilities for rolling certain sums with three dice.

- **Calculating Probability for Simple Target (12)**:
    - For a target of 12, the possible outcomes (12, 13, 14, 15, 16, 17, 18) and their probabilities are calculated based on the combinations possible with three dice:
        - **Probability Calculation**:

          Total possible outcomes = 6 * 6 * 6 = 216 (for three six-sided dice).

          Number of outcomes that sum to 12 or higher = 73.

          Probability of achieving a sum of 12 or higher: $73/216 \approx 0.338$ or 33.8%.
- **Calculating Probability for Difficult Target (5)**:
    - For a target of 5, possible sums are (3, 4, 5). These sums have fewer possible combinations than a higher target.
        - **Probability Calculation**:

          Total possible outcomes = 216.

          Number of outcomes that sum to 5 or lower = 10.

          Probability of achieving a sum of 5 or lower: $10/216 \approx 0.046$ or 4.6%.

Thus, achieving a sum of 5 is more challenging (a lower probability) than achieving a sum of 12.

Step 3: Implementing the Consensus Algorithm

Here consensus is achieved probabilistically based on dice rolls.

- **Roll Dice**: Each participant rolls three dice, yielding a random sum between 3 and 18.
- **Target Comparison**: If the sum meets or exceeds the target, the participant can propose a new block.
- **Consensus**: If multiple participants succeed simultaneously, the highest roll wins. The pseudo-code below demonstrates the algorithm logic:

```
import random

def roll_dice():
  return random.randint(1, 6) + random.randint(1, 6) + random.randint(1, 6)

def consensus_algorithm(target):
  while True:
    roll = roll_dice()
    if roll >= target:
      print(f"Consensus achieved with roll {roll}")
      return roll
```

Step 4: Decentralized Consensus Verification

Once a participant achieves consensus by matching or exceeding the target, other nodes in the network verify the roll by checking:

- **Roll Validity**: Verifying the randomness and outcome of the dice roll.
- **Compliance with Target**: Ensuring the roll matches or exceeds the target for that round.

Upon verification, the block is added to the blockchain, and the consensus is considered reached.

Comparisons with Proof-of-Work (PoW)

In a PoW system, consensus relies on computing a hash below a specific target. The Three Dice Consensus Algorithm simplifies this by using probabilities and a predefined target to achieve consensus, similar to PoW but with less computational intensity.

Summary

The Three Dice Decentralized Consensus Algorithm provides an innovative probabilistic approach to blockchain consensus. By setting targets and calculating probabilities, participants achieve consensus without centralized oversight. This algorithm demonstrates that blockchain consensus can be achieved by chance, aligning with principles of decentralized and trustless environments.