1 Blockchain

Lists and blocks

Given a set S, let $\textsc{Set}(S)$ be the set of sets of elements of S and $\textsc{FList}(S)$ be the set of all finite lists of elements of S. Given $L \in \textsc{FList}(S)$, we use notation $\text{length}(L)$ to refer to the number of elements in L, notation $L[i]$ to refer to the i-th element in L, where $i \in \{1, \ldots, \text{length}(L)\}$, and notation $L[i, j]$ to refer to the sublist $[L[i], \ldots, L[j]]$ of L, where $i, j \in \{1, \ldots, \text{length}(L)\}$ and $i \leq j$. Notice that $\text{length}(L) = 0$ if and only if L is the empty list $[\,]$. Finally, we say that a list L_1 is a prefix of a list L_2 if L_1 is the empty list, or $1 \leq \text{length}(L_1) \leq \text{length}(L_2)$ and $L_1 = L_2[1, \text{length}(L_1)]$.

We assume a fixed alphabet Σ, and that blocks contain the following information:

- A word in Σ^* that serves as the id of the block.
- The id of the previous block.
- Another word serving as an identifier of the owner of the block.

Body of Knowledge and Blockchain

+++ explain how blockchain works, nodes have ids and previous block hash so one mines blocks after other blocks +++

Let us fix a *genesis block*, identified as ϵ and that does not contain neither a previous block id or an owner id.

We represent all knowledge in a blockchain system as a tree K rooted in the genesis block: the set of nodes of of this tree are all blocks that have been mined and there is an edge from block b_1 to block b_2 if b_2 was mined from b_1. We denote these trees as *body of knowledges*.

+++ explain the notion of the blockchain ++++

Given a body of knowledge K, we say that the blockchain of K is the list formed from the longest path from the root to the tree, if such a path is unique. If two or more different paths are tied for the longest, then we say that the blockchain in K does not exists. We use the notation $\text{bchain}(K)$ as a function that returns the blockchain of K, if it exists, or the empty list otherwise.

2 Mining game

+++ explain a bit on how blockchain uses miners to validate, and that whomever wins a block receives a mining reward. Include the notion that our blocks +++

The mining game is played by a set $\mathcal{P} = \{1, \ldots, m\}$ of players, and the set of states of our game consists of all possible body of knowledges in which all blocks except for the genesis block are owned by one of the players in $\mathcal{P}$.

On each step, miners looking to maximise their rewards choose a block in the current body of knowledge, and attempt to mine from this block. Thus, in each turn, each of the players race to put the next block in the body of knowledge, and only one of them succeeds. The probability of succeeding is directly related to the comparative amount of hash power available to this player, the more hash power the likely it is that she will mine the next block before the rest of the players. Once a player puts a block, this block is added to the current state, obtaining a different body of knowedge, and the game continues from this new state.

In order to formally define our game, let us denote by $(\mathcal{P})$ the set of all possible blocks owned by any player in $\mathcal{P}$, and $\text{BK}(\mathcal{P})$ the set of all body of knowledges constructed from blocks in $(\mathcal{P})$, so that the states of our game are precisely $\text{BK}(\mathcal{P})$.

A strategy for a player p is a function $\text{BK}(\mathcal{P}) \to (\mathcal{P})$ that assigns to each body of knowledge a block of it where player p wishes to mine next.