DeCI - Decentralized Computation Infrastructure

Team: Jenny from the block(chain)
João Correia SCIPER 343955
Georgios Fotiadis SCIPER 271875

Workload

João

- Estimating the cost
- Gathering available nodes
- Distributing the data and executables

Georgios

- Joining the network
- Updating node's balances
- System performance analysis

Workflow - user perspective

Georgios

1. API and simple command line interface
2. Input executable and list of inputs.
3. Define number of nodes desired to distribute the workload.
4. Collect results and pay for the computations.

Workflow - internal phases

Georgios

0. Joining the network - George
1. Cost estimation - João
2. Node aggregation - João
3. Workload Distribution - João
4. Budget update - George

Joining the network

Georgios

- Public nodes with known IPs
- Entering nodes add a public node as a peer and broadcast special Join message
- Other nodes reply with an AckJoin message and add entering node as a peer
- Entering nodes block until they collect enough AckJoins

Cost estimation

João

- Measure the execution time of 1 to 3 random inputs;
- Cost per unit: average number of seconds it takes to execute one input (decimal);
- There’s also a base fee of 1 “coin” per requested node;
- Total cost: **number of inputs * cost per input + 1 * number of requested nodes**

Node Aggregation

João

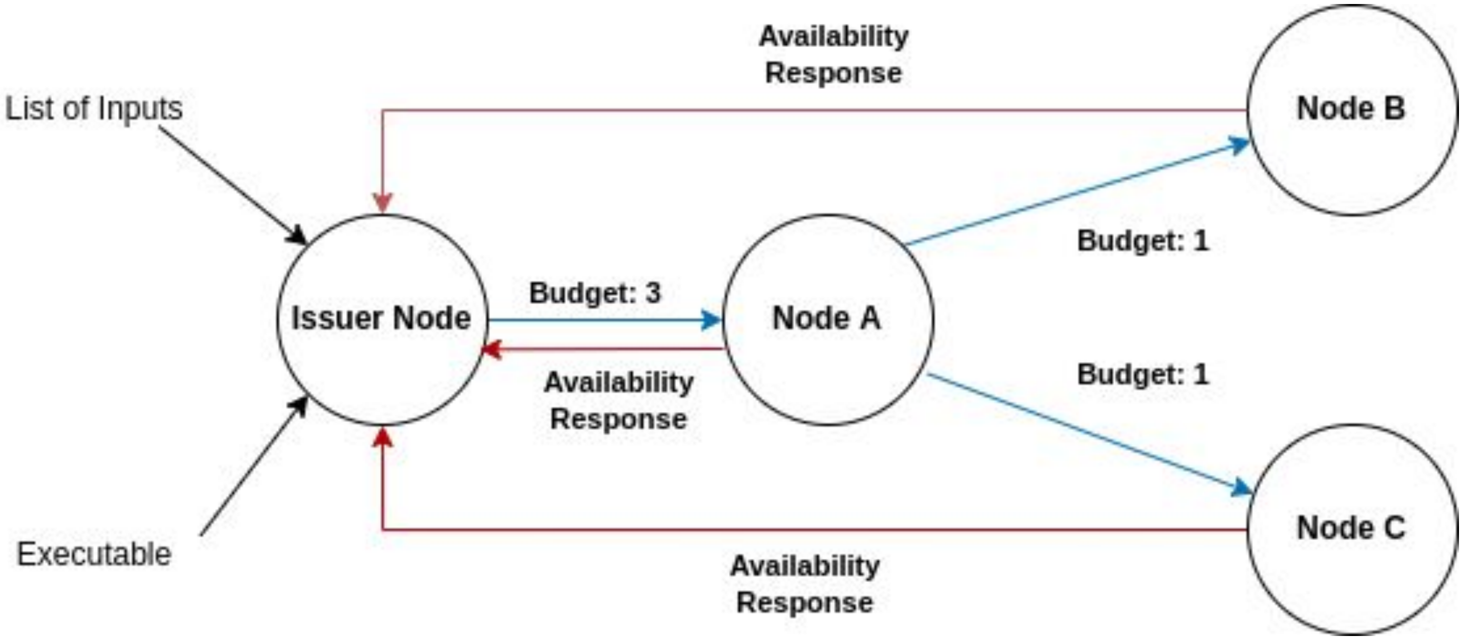
Objective: gather enough nodes to distribute the workload as specified by the user.

AvailabilityQueryMsg: used to ask nodes to participate in a given computation.

Behaviour inspired by budgeted search requests of hw 2.

Node Aggregation

João



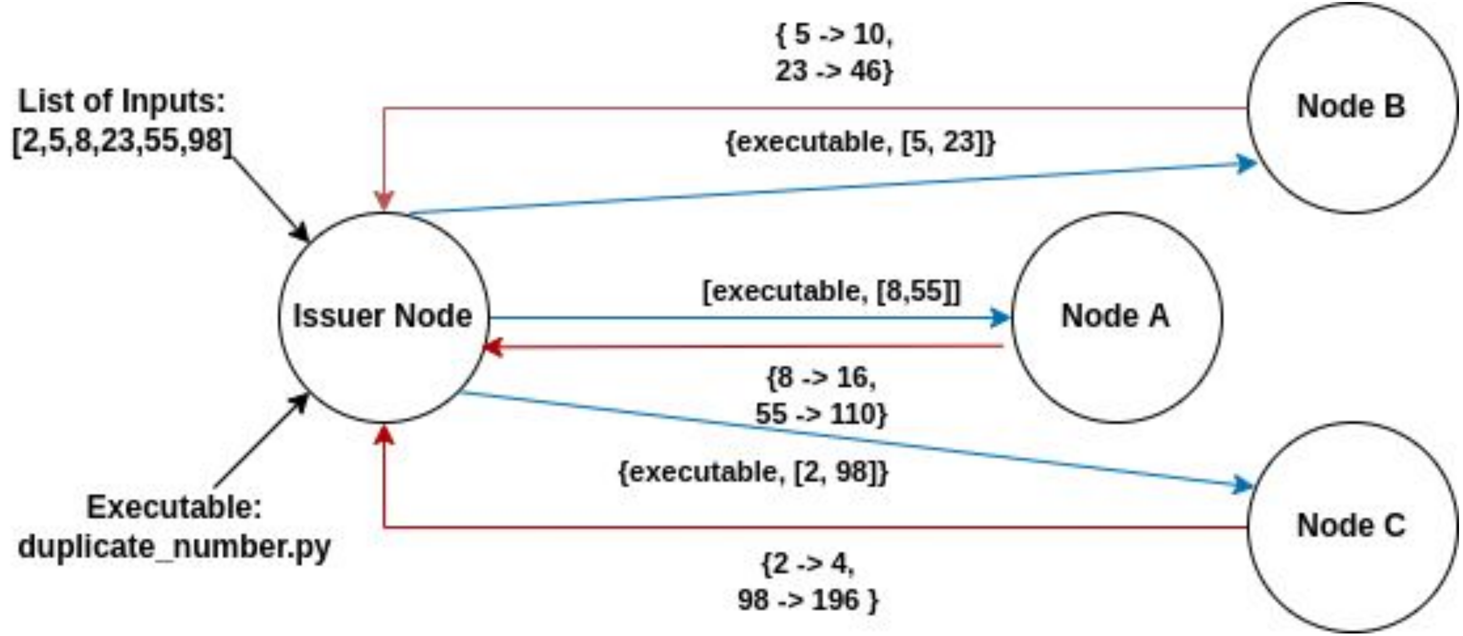
Workload distribution

João

- List of inputs is distributed evenly among the available nodes.
- The executable is sent as a byte stream along with some metadata and instructions to run.
- Remote peers save the byte stream as an executable and record the outputs created by the input list, sending them to source.

Workload distribution

João



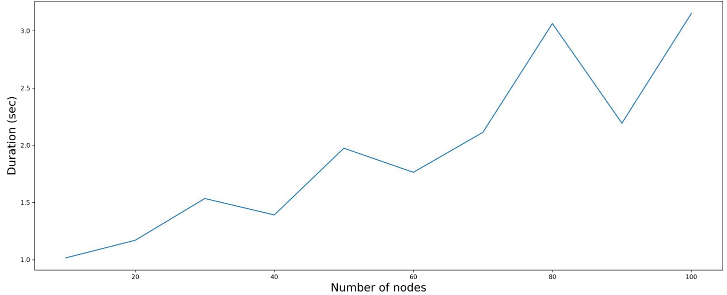
Budget Update

Georgios

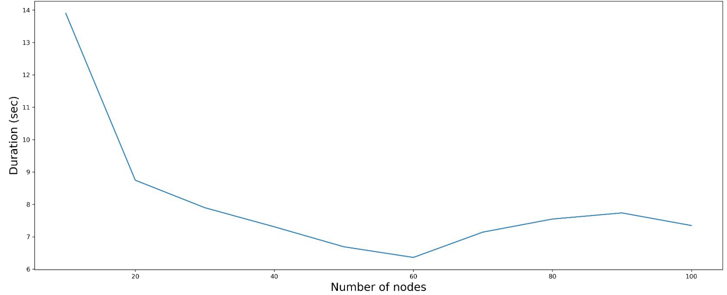
- Blockchain based solution to keep track of transactions
- Issuer node removes the calculated amount from its budget and broadcasts a special message with a budget map
  - Map from node address to money earned
- Receiving nodes check if their address appears in the map
  - Update budget or ignore
- All nodes record the transaction on a blockchain

System Performance Analysis

Georgios



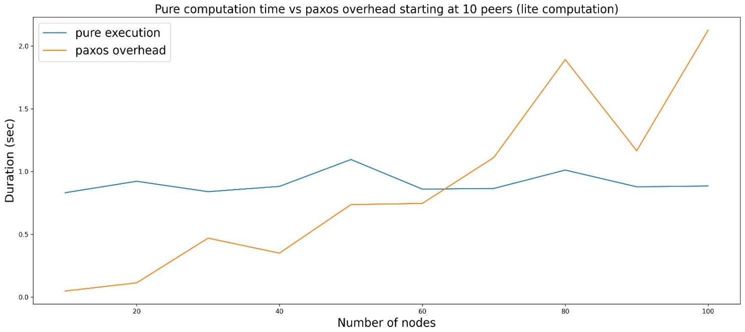
Computation duration evolution with increasing number of nodes starting at 10 peers (lite computation)

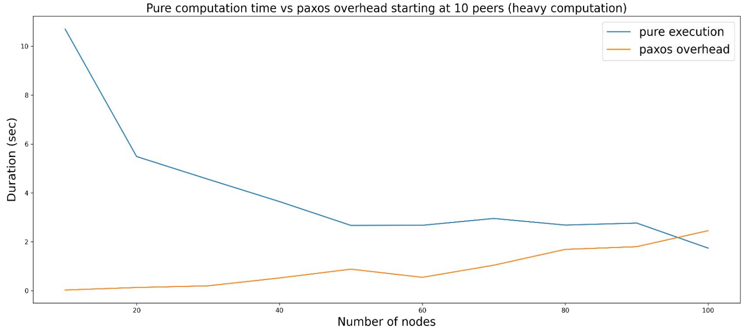


Computation duration evolution with increasing number of nodes starting at 10 peers (heavy computation)

System Performance Analysis

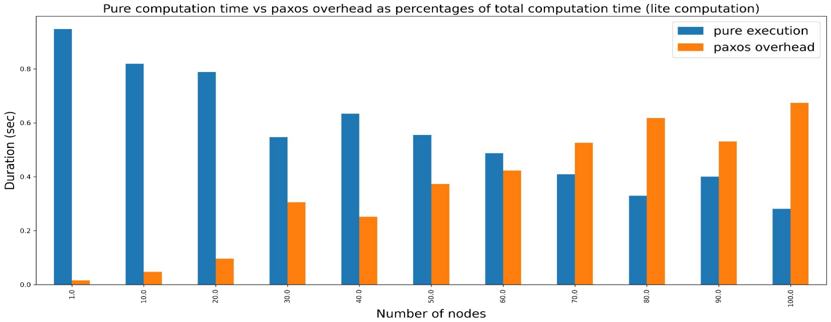
Georgios



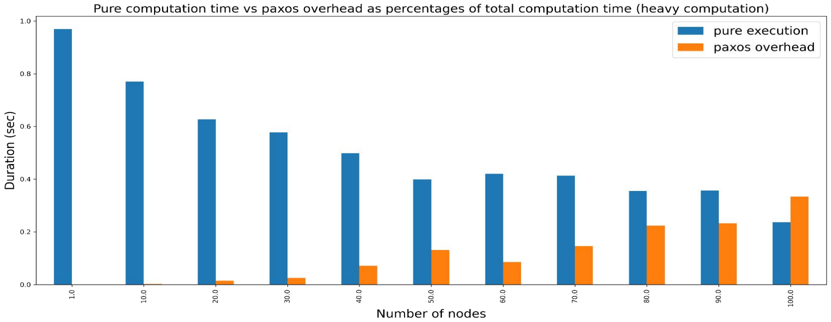


System Performance Analysis

Georgios



Pure computation time vs paxos overhead as percentages of total computation time (lite computation)



Pure computation time vs paxos overhead as percentages of total computation time (heavy computation)

On to the demo!