



Sudoku Solver

Computação Distribuída 2023/2024

Giovanni Santos 115137

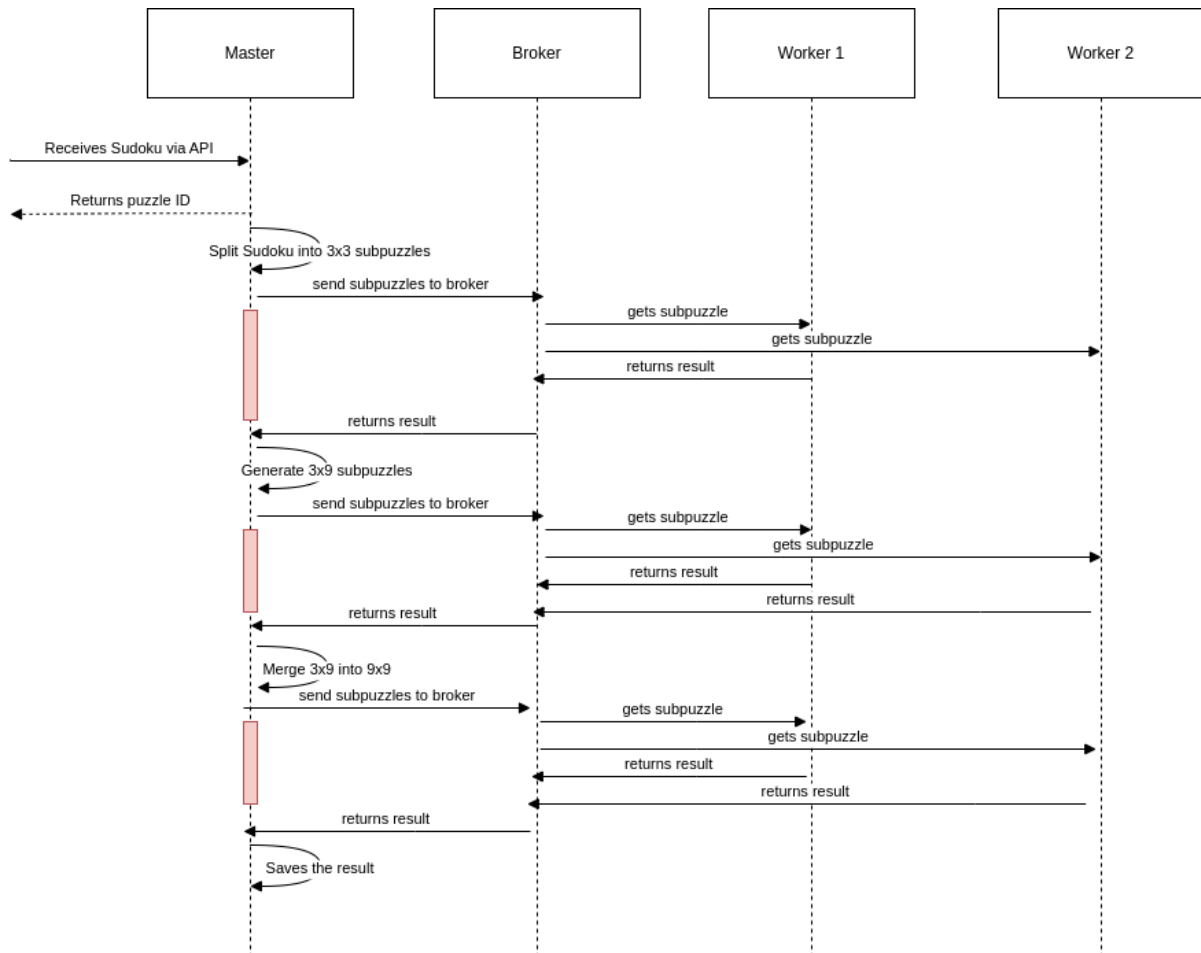
Edivaldo Gustavo Castro Luís Bonfim 110124

O trabalho tem como objectivo construir um sistema distribuído capaz de resolver um puzzle Sudoku. O sistema tem uma arquitetura baseada no algoritmo Map-Reduce, contendo um broker centralizado e um conjunto de processos centralizados(workers).

O projeto consiste em duas APIs principais `/solve` a onde o servidor recebe o puzzle para resolver e retorna um ID relativamente ao puzzle, e `/solution/<puzzle-id>` que recebe o ID de um referido puzzle e retorna o puzzle se o puzzle já estiver resolvido, ou o número de puzzles resolvido até momento caso o puzzle referente ao ID ainda está a ser resolvido.

- Master / API Server: É o componente principal que recebe as requisições do cliente e lida com a lógica de divisão de tarefas e redução das soluções,. De uma forma mais descritiva o master:
 - Receber requisições,
 - Gerar sub-puzzles e enviar para o broker,
 - Criar subpuzzles 3x9,
 - Criar puzzles 9x9.
- Workers: Recebem e processam as tarefas, sendo elas:
 - Gerar todas as configurações para um dado subpuzzle,
 - Validar puzzles 3x9,
 - Validar puzzles 9x9.
- Celery: Sistema de fila de tarefas que permite a execução distribuída, usado para enviar mensagens
- Rabbitmq: Broker para enviar as tarefas entre o master e os workers, garantindo a comunicação assíncrona entre eles.

Protocollo:

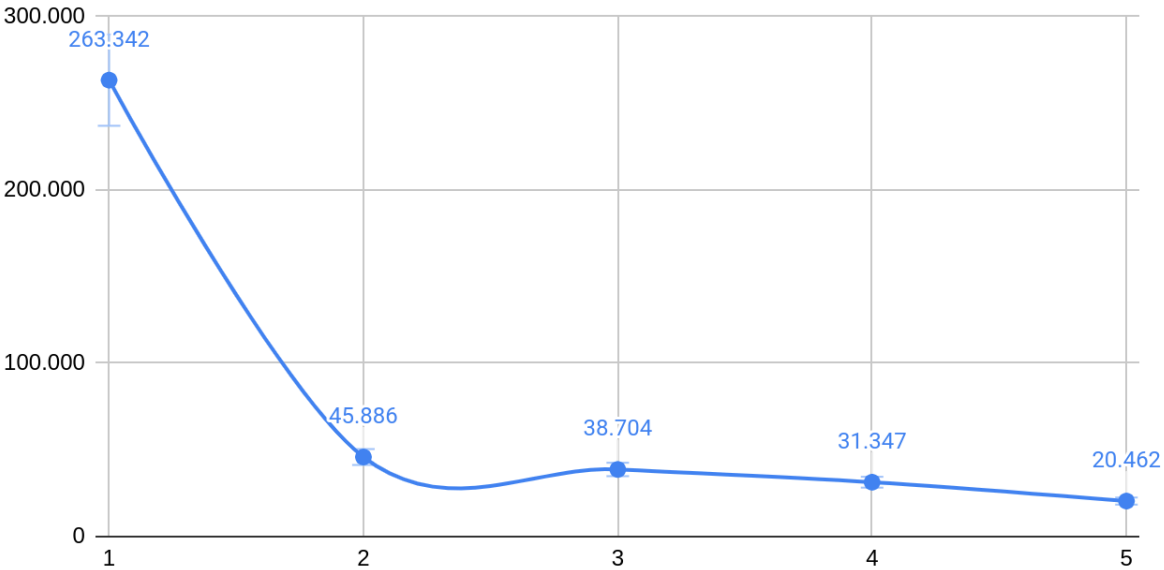


Puzzle exemplar

[illegible]

Number of Workers	Time to Solve (s)
1	263.342
2	45.886
3	38.704
4	31.347
5	20.462

Relação workers tempo



Conclusão

A curva mostra uma tendência de diminuição no tempo de execução conforme o número e workers aumenta. Mas a eficiência de paralelismo diminui à medida que mais workers são adicionados. Isso devido à sobrecarga de gerenciamento de comunicação entre os workers.