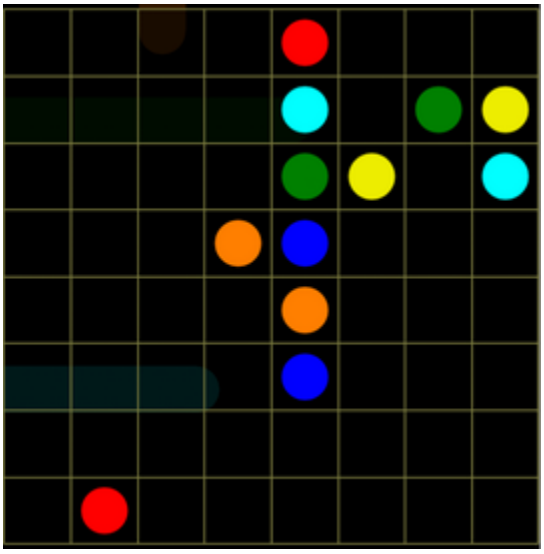
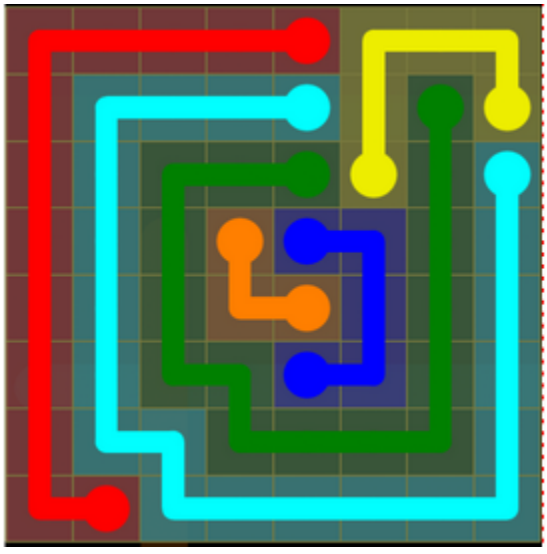
Parallel Flow Solver

Deji Oyerinde(oko2107) , Kidus Mulu(km3533)

The Problem

Flow free is a video game that is played on a grid. Some of the boxes of the grid have dots of different colors. Each colored dot has a corresponding second dot somewhere else in the grid, and the goal is to connect all the pairs of similar colored dots without crossing any lines.

For example, a possible starting grid is shown in the following table along with a solution:

Start grid	Solution
	

Along with getting a correct solution with all the same-colored pairs connected without crossing any paths, the original game also keeps track of the number of steps that were taken to finish the game, which is the number of times a connection was formed between two dots, and an optimal solution is one that connects all the pairs in the minimum number of steps.

The Solution

There are two possible approaches : Formatting the problem as a Constraint Satisfaction Problem or performing Shortest paths. The Shortest Path approach involves using an informed, best first search such as A*. The heuristic would be (the number of empty spaces) + (number of moves done at a particular state). The parallelization would be implemented in the selection of the possible successor states from a single state.

As for the CSP approach, we'll reduce the problem to a SAT problem and represent the constraints for the board as a CNF.

We'll experiment with both approaches and choose the more optimal/feasible solution.

Backup Plan

If implementing the solution for flow free proves to be too difficult, our plan B is to implement Johnson's algorithm for finding the shortest path between all pairs of points in a graph, which is a mix of the Bellman-Ford algorithm for ensuring there are no negative cycles and also doing transformations on the graph and then applying Dijkstra's algorithm to the nodes. The parallelization would be implemented in the internals of the Bellman-Ford and Dijkstra so that adjacent nodes are examined concurrently.

References:

- https://mzucker.github.io/2016/09/02/eating-sat-flavored-crow.html
- https://mzucker.github.io/2016/08/28/flow-solver.html