Programming Project (optional)

due: Sunday, January 17th, 2021, 23:59

The assignment is Coordinated Motion Planning as described in the Computational Geometry Challenge (Challenge, for short) 2021: `https://cgshop.ibr.cs.tu-bs.de/competition/cg-shop-2021/`.

Given a set of n axis-aligned unit-square robots in the plane, a set $S = \{s_1, \ldots, s_n\}$ of n distinct start pixels (unit squares) of the integer grid, and a set $T = \{t_1, \ldots, t_n\}$ of n distinct target pixels of the integer grid. During each unit of time, each robot can move (at unit speed) in a direction (north, south, east or west) to an adjacent pixel, provided the robot remains disjoint from all other robots during the motions. This condition has to be satisfied at all times, not just when robots are at pixel positions. For example, if there are robots at each of the two adjacent pixels (x, y) and $(x + 1, y)$, then the robot at (x, y) can move east into position $(x + 1, y)$ only if the robot at $(x + 1, y)$ moves east at the same time, so that the two robots remain in contact, during the movement, but never overlap.

In addition, for some instances, there may be given a set of obstacles, consisting of a number of stationary, blocked pixels that cannot be used by robots at any time.

The task is to compute a set of feasible trajectories for all n robots, with the trajectory for robot i moving it from s_i to t_i. The Challenge is run in two categories, with the two following objective functions:

- minimize the makespan, i.e., the time until all robots have reached their destinations;
- minimize the total distance traveled by all robots.

You can choose to focus on one objective function only.

You will find ample input examples on the Challenge's website. We will use the same input and output type and format as in the Challenge.

Your submission should include (i) a description of your algorithm(s) and their implementation, (ii) a discussion of the performance of your solution on a few (three is a good number) types of inputs, and (iii) all the (documented) source code necessary to generate an executable of your program, together with generation instructions. You may write the program in `C/C++`, `Java` or `Python`.

You may submit this assignment in pairs—you are encouraged to do that. Feel free to discuss the project with your fellow students. However, if you wish to share code across projects, you need to acquire the permission of the course's team as well as to state this explicitly and in detail in your submission.

Additional information about the project, as needed, will be posted on the course's website. Stay tuned.