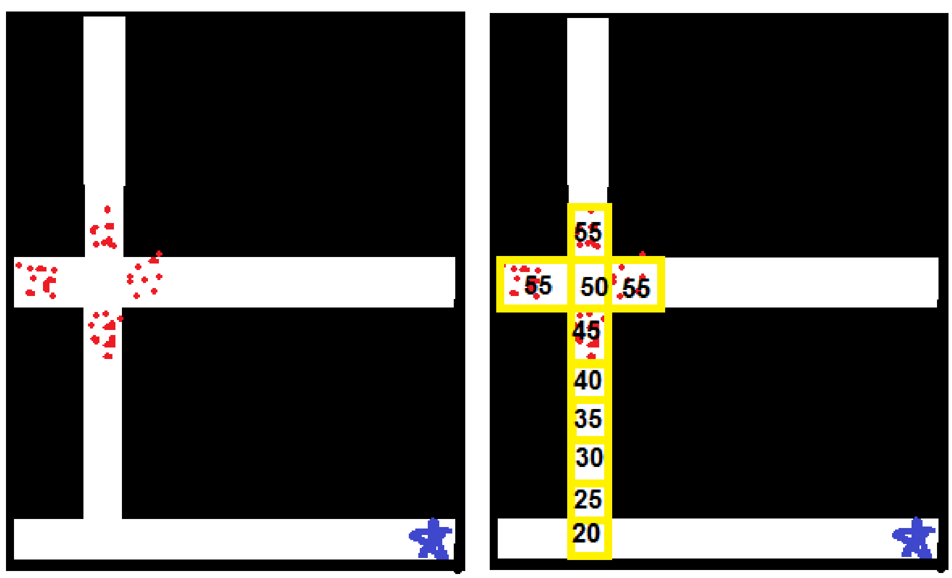
1. Problem formulation: aggregate particle swarms at specified location under global input:
   (1) Deterministic method: specifically, this method requires a gradient-like function based on
      (a) particle swarm locations (RRT cost and Euclidean coordinates)
      (b) particle swarm density (large density group has more weights/priority)
      (c) swarm homogeneity (number of different swarms, the smaller the number is, the higher the homogeneity will be, and thus the higher of control efficiency)
      (d) node-wise relationship (close relatives, not pairs close to each other in Euclidean sense, can be aggregated first to increase homogeneity)

      And such function should specify a unique solution given an input of particle locations and density distribution.

      The difficulty lies in the function construction and a unique solution. For example, in fig. 1, solution A is to move swarms into the vertical channel and aggregate them at the bottom, and then move to the target place. This would take about **11 steps** before aggregation (d,d,l,d,r,d,d,d,d,d,d), and total **cost of 75** to the target place. Solution B is to aggregate all swarms at the left corner first, and this takes about **7 steps** (r,r,d,r,u,u,l). And the total **cost to the target is 90**. At the first glance solution A is absolutely better. However, this only considers one region of interest, where solution A requires 11 successive operations, and solution B requires 7 successive operation. Therefore solution A might not be fair for other swarms, say there are 10 other swarms, although locally it gives an efficient solution. Solution B only takes 7 steps before aggregation, which increases homogeneity (number of different data groups, i.e. swarms) faster than A and hence it is a globally preferred solution.



   (2) Randomized method: randomly distributed control input with adaptive priority range. This is effective at the beginning, or specifically, the moment with low data homogeneity, namely, when there is large number of different swarms (# > 50), random input helps

decrease such data variation and increase homogeneity. And with small number of swarms, we can move them around more efficient. And one apparent advantage of randomized method is no computation cost at all to make a decision

2. Read and gave comments on Xin's paper.