Ning

30 September 2019

Metaheuristics

Class Summary

[OPENING]: Spending more time fleshing out what a metaheuristic is suppose to do and how it helps us solve the problems that we are concerned about, as well as diving into the preliminaries of Tabu Search.

[SUMMARY]: Neighborhood is the space explored by visiting points within the neighbors of the chosen point, and then the progression continues as such. The benefit is that it limits the subsets of potentials in order for our selection choice to be better. A metaheuristic is defined by its neighborhood, and a search operation. The more agreeable way to define a neighborhood is through a set of transformations (or movements) rather than an exhaustive list. ie: the North South East West transformation can be represented by (i,j) indications, and +1/-1 accordingly to the i/j index value to simulate NSEW.

[SUMMARY]: A fitness landscape is a representation of fitness function vs elements in S. We care about whether the landscape is simple vs difficult. Here, simple or difficult is dependent on how smooth/rough f varies as we move along the neighborhood. The efficiency of the metaheuristic depends on the neighborhood. A population-based metaheuristic differs from the type of single-point based approach we’ve been looking at for a solution in S. Population-based means considering a neighborhood/population of points in order to obtain a solution to S. We have a trajectory of population to consider. The solution is the best individual amongst all the population generated. The genetic algorithm is an example of this type of population-based approach. This fits the same formalism we’ve discussed, except that the search space is (note the neighborhood is calculated using all values from the previous neighborhood, meaning that generating neighbors is a tougher problem). Defining neighbor is done via a rule to generate one successor within that neighborhood rather than an exhaustive list (like we did in our project).

[SUMMARY]: Tabu Search. This metaheuristic performs well for Quadratic Assignment Problems (QAP). We can generalize the traveling salesman problem as a QAP. Progression is consistent with what we’re familiar with, the only change being how the selection operation chooses the point amongst neighbors. The premise of the tabu search is that it allows you to explore previously unexplored areas of your search space by allowing you to choose sub-optimal selections, under the constraint that your choice is not in the tabu list (the existence of forbidden choices). Typically, this list is used to prevent cycling through a set of solutions that you’ve already explored. This list is constantly updated. Tabus should not be permanent. The flowchart for the tabu search is as follows:

1. Generate an initial solution (usually randomly)

2. Create an empty tabu list.

3. Generate the neighborhood.

4. Choose the best non-tabu neighbor.

5. Update the best solution seen so far.

6. Check whether the end condition is met.

6a. If the end condition is met, output the best solution seen so far.

6b. If the end condition is not met, go to step 7.

7. Update the tabu list with a movement ()

8. Proceed back to step 3 (generate neighborhood).

[SUMMARY]: The tabu list memory size is the value that essentially determines how far you can explore within your neighborhood before you find another point of improvement (step-wise). Dead-ends can also occur (in the NSEW example, all 4 paths are blocked). If your memory is too big, then you could get stuck early. Usually, what values are in the tabu list? 1. Already visited points (rarely done in practice). 2. Attribute of the solution seen previously (i.e: value of fitness). Generally, neighbors are defined through movement, so we can also restrict movements (neighbors) that we think are forbidden. (i.e: tabu movement could restrict the direction you came from in case of NSEW). In a permutation space, movements are specific but also symmetric, we can just restrict the movement (the swap). Management of the list needs to happen frequently, we can’t have permanent items on the list, and they should help us avoid blockage.