July 14, 2021

ECE 457A: Assignment 3

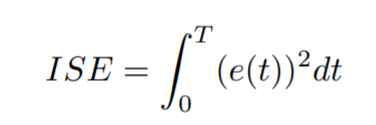
Daivik Goel, Bosco Han, Lichen Ma

# Question 1

### Suitable representation:

### Formulate a fitness function used to evaluate a solution

The fitness function used to evaluate a solution is the inverse of the integral squared error (ISE):



Where e(t) is the error signal in time domain.

Thus, to optimize the PID controller, we need to minimize the ISE value.

### Genetic Algorithm with population of 50, generations = 150, crossover probability = 0.6, mutation probability = 0.25

### with fitness proportionate selection and elitism survival selection to keep the best two individuals

### Plot of fitness of best solution in each generation

# Question 2

# Question 3

Using the NetLogo web interface we get to see the ANT model in action. For this question we will try varying the population size, diffusion rate and evaporation rate to see what effect it has on the time it takes to get the food from the piles.

Chart, line chart

Description automatically generatedThis is our base case with population set to 30, diffusion rate set to 40 and evaporation set to 10.

In increasing our population to 50 and keeping the rest of the parameters the same we see longer time for pile 2 and 3 to be destroyed. In fact, it almost looks like the time increase is directly proportional to the increase in population. As the population went up 1.6x the time for the piles to be finished also seemed to go up the same amount.

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generatedIn increasing our population to 100 and keeping the rest of the parameters the same we get the following graph. The time for each of the piles drastically decreases.

Chart, line chart

Description automatically generatedIn increasing the diffusion rate to 80 while having a 100 population and the evaporation to 10 like was previously we get the following graph.

With the diffusion rate increased we see the food in the piles overall decrease quicker with the exception pile 2 which takes longer.

Chart, line chart

Description automatically generatedIncreasing the evaporation rate from 10 to 20 while keeping all other parameters the same gives us the following graph.

The graph resembles the previous one but piles 3 takes around 3 times the amount of time to be fully eaten.

Below are the results from the code implementation. The code itself can be found in the attached repo and it has a readme for your convenience. These are the parameters we are using in the base case. We change them as required in the different parts.

alpha,beta,base\_ph ant\_pop,phermone\_decay,state\_transition,online\_phermone,Q,iter =

1,1,1,10,0.4,0.5,True, 5000,200

Chart, histogram

Description automatically generatedFor our base case with the values seen above we got this as our graph:

As you can see, as the iterations increase, the found distance also massively decreases until we reach a point of small variations. This is probably due to the increasing pheromones highlighting the shortest path for future iterations.

Chart, histogram

Description automatically generatedWhen we change the values of the Pheromone Persistence, we get the following graph:

All the graphs reach the same point of small variations, but we see that a higher pheromone persistence reaches it much faster. This makes sense given a higher persistence should lead to more ants following previously found shorter routes.

When we change the state transition parameter, we get the following:

Chart

Description automatically generated with medium confidence

In our code a smaller state transition values favors a pheromone-based city selection rather than probability. We see a drastic difference as the one that chooses solely based on the highest pheromone stays constantly at the top of the distance. This seems to indicate that the a certain level of randomness in the ant’s selection is required to get the best solution possible.

Chart

Description automatically generatedWhen we change the population size of the ants, we get the following graph:

This graph shows us that the slope of the curves to get to each population respective resting points is similar. However, the rest points themselves are greatly affected especially with the 5 ants resting almost 2000-3000 more distance than the 10 and 15 population ants. It seems that population of ants are required to reach a certain value which once reach renders the impact of the difference in ants to be minimal.

Chart, line chart, histogram

Description automatically generatedFinally, when switching between offline updating and online delayed updating we get the following graph.

Although the slopes to their respective resting points are similar, we see that offline updating is quite lower consistently than online updating. We also see less variation in the offline updating line while we have a lot more variation with online updating. This seems to make sense as with offline updating since we are only updating the pheromone of the best ant in the iteration, the paths that the ants take in future iterations should and do have a distinct advantage. We see this reflected in the graph above.