UNIVERSITY OF
LOUISVILLE
J. B. SPEED SCHOOL
OF ENGINEERING

Computer Engineering and Computer Science
Department

# CECS545-Artificial Intelligence

## Project 4

Dr. Roman Yampolskiy

# Project 4: TSP – Genetic Algorithm

◆ Learning objectives. At the completion of this project, you should be able to:

- Implement a genetic algorithm for an intractable problem.

- Be able to discuss aspects of the GA that control success in finding good solutions.

# Problem

- You will be expected to implement a GA for a TSP dataset with up to 100 cities
- Data for each problem will be supplied

# Hints

- Analyze the results of two different settings for two different parameters. For example you can use two different types of crossover methods and two different mutation rates. In that case, you need to collect four sets of data:

|            | Crossover A | Crossover B |
|------------|-------------|-------------|
| Mutation 1 | Dataset 1A  | Dataset 1B  |
| Mutation 2 | Dataset 2A  | Dataset 2B  |

# Hints

- For each of the four datasets, you will need to run your code several times to develop performance statistics.

- I use the example of two crossovers and two mutations. You can select any modification of GA so long as you can have (at least) 2 options for each modification. For example you could choose two different sizes of populations. You could choose two different selection methods, etc.

- Don't forget, if you run your code a 100 times on a large dataset, you will likely wind up with nearly 100 different solutions.

# Deliverables

- Well-commented source code for your project. You can use any language you like, but I reserve the right to ask you to demo performance of your algorithm on a new dataset.

- Include a GUI with visual (and dynamic) representation of the solutions for this project.

- Project report (3-4 pages).

# Deliverables

- Define your crossover and mutation methods. Or else define whichever two GA elements you chose to investigate. Clearly identify if your methods were your idea or else the source from which you got the idea.

- Define your stopping criteria.

- Clearly identify your best solution for the solved problem.

- Analyze the effectiveness of the two chosen variations in GA. To do this you will likely need to run your code several times and provide the min, max, average and standard deviation of each set.

# Deliverables

- Briefly discuss how long a typical run took for each of the datasets.
- Graphically present your improvement curves.
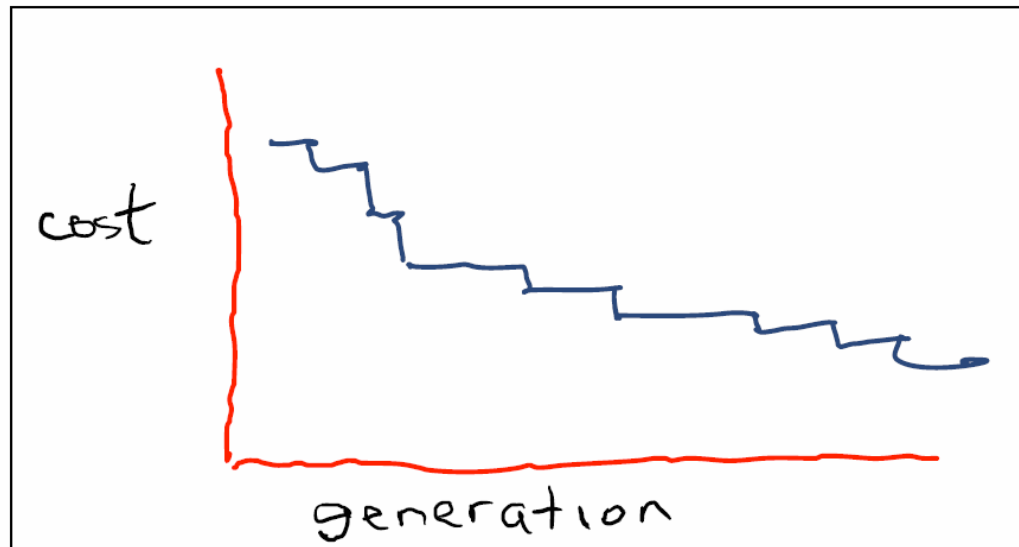
# Critical Thinking

- Describe the biggest problem that you had in the implementation and analysis of this project.

- Describe the one thing you would change if you did this project again.

- Please elaborate on what you learned from using GA. At least one paragraph, please.

- Give your overall impressions of GA as problem solving technique.