# GATSPE: A Generative Adversarial TSP solver based on Evolutionary Algorithms

Guo Wei, Li Zhaofeng, Jia Baoqi
School of Computing, National University of Singapore

## Motivation

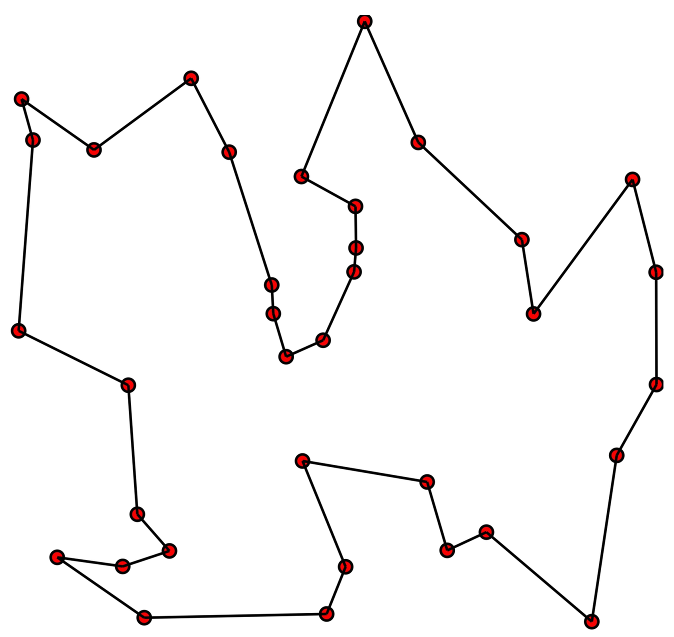
- **Evolutionary Algorithm (EA)** is a family of search algorithms inspired by biological evolution. EAs are a **powerful tool for optimization problems** that are difficult or impossible to solve using traditional optimization techniques, including NP-hard problems.
- However, the searching process of EA at each optimization step depends on some **randomness** which may not ensure an optimized end result.
- We propose GATSPE, to incorporate **deep learning methods** to improve the optimization at each step but remain overall computational efficiency and solution quality. So that both the efficiency and solution quality can be improved compared to randomness-oriented EA algorithms.

## Objective

- Our project aims to use **Travelling Salesman Problem (TSP)** as a cut point to explore the utilization of **deep learning methods** to improve the efficiency and solution quality compared to **randomness-oriented** EA algorithms.

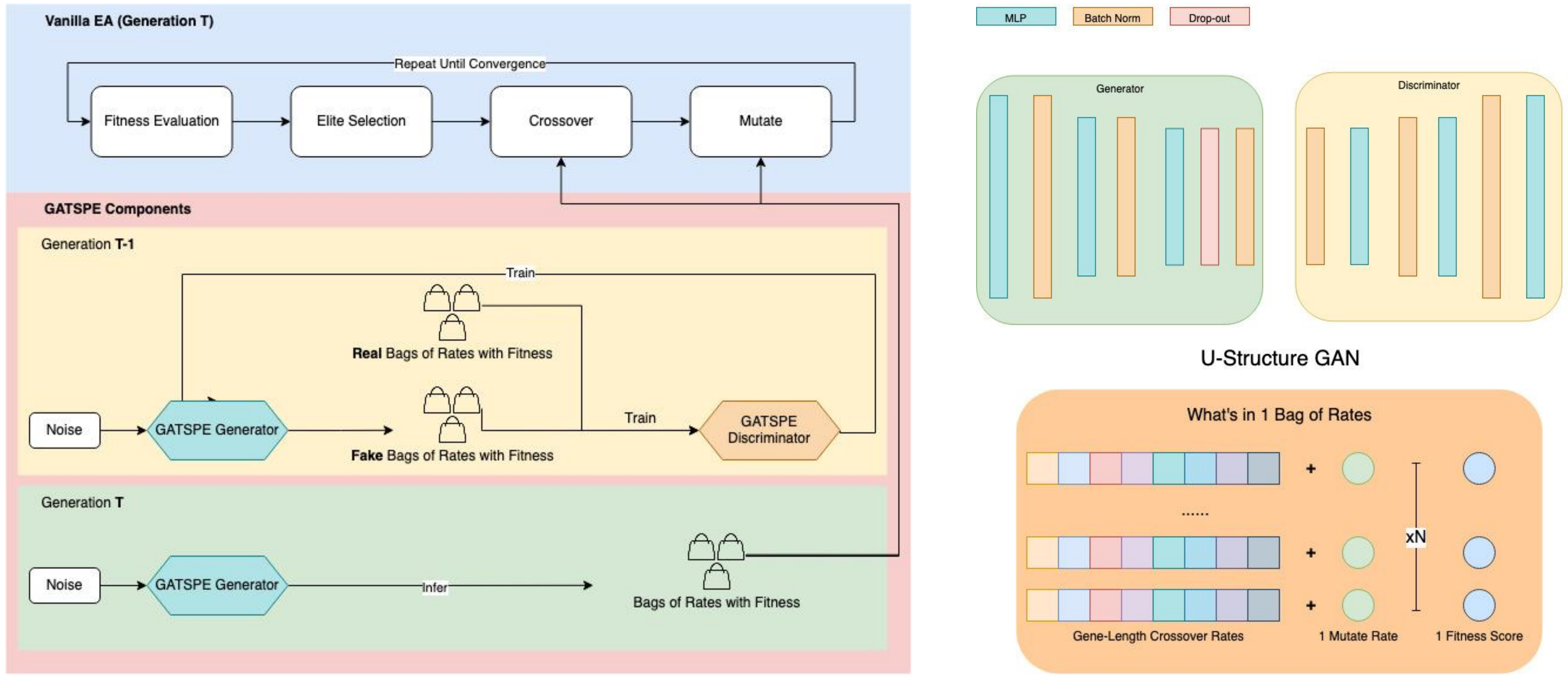
## Problem Formulation

- **TSP** is defined as: *Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city and returns to the origin city?*



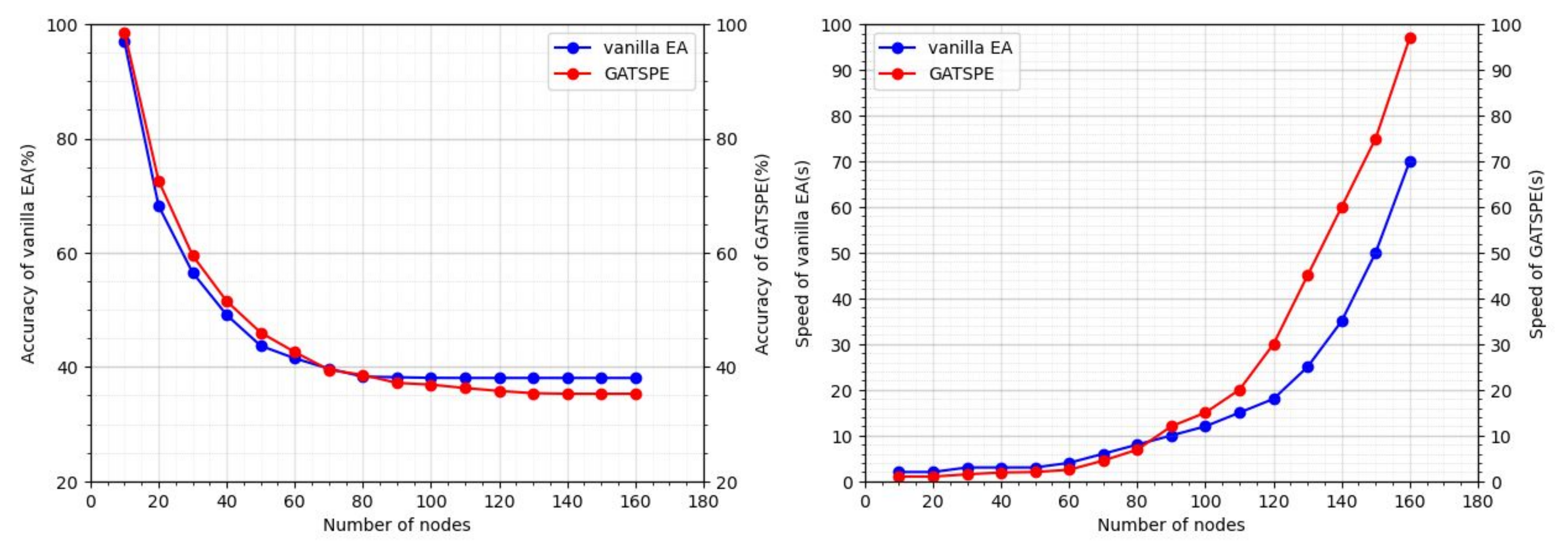
- **EA** algorithm solves **TSP** by by generating and evaluating candidate solutions and iteratively improving them over time.
- Candidate pairs are typically generated using the genetic operators of crossover and mutation. **Crossover** combines two parent solutions from the population and creating an offspring solution by exchanging some genetic material between them. **Mutation** introduces small random changes to the genetic material of a single solution.
- During this process, the amount of change is governed by crossover rate and mutation rate, which are hyperparameters to be defined in EA algorithm.
- Our approach is to iteratively train a **GAN** alongside the EA algorithm to generate crossover rate and mutation rate, aiming to produce more elite candidates by the generated rates.

## Model Architecture




## Experiments

- We performed experiments on TSP problem ranging from 10 nodes to 160 nodes.
- We compared the accuracy and inference speed of vanilla and our GATSPE as performance metrics.




- As can be seen from the charts above, GATSPE performs faster and with better accuracy than vanilla evolutionary algorithms on TSP problem when the number of nodes are kept within in a reasonable range.
- However, when there are too many nodes on the graph, which means too many cities for the salesman to travel, GATSPE may perform slower due to the increase of training time, but it still gives comparable testing performance.

## Training & Inference

- **Asynchronous Online Training:** To capture the long term trend of best-performing crossover and mutation rates, the model is **trained every n generations** from the EA loop. The training data is accumulated over time, with an identifier to differentiate generations.
- **Zero-Elimination Inference:** The GAN model can output 0 crossover and mutation rates for some positions on a gene. However, since EA heavily depends on random searching for optimization, we **add an epsilon** (1e-4) to the zero probabilities.

## Conclusion and Future Work

- GATSPE can effectively learn to generate crossover and mutation rates, leading to improved performance. It also helps to optimize the trade-off between exploration and exploitation in evolutionary algorithms.
- However, there are several challenges such as loss function design, overfitting, and the computational cost.
- Future directions include to generate other types of hyperparameters such as the population size or selection strategy, and to improve the scalability and generalizability to new problems.