NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

School of Computer Science and Engineering
College of Engineering

Efficient Graph Neural Networks

for Travelling Salesman Problem using Multilevel Clustering

Student: Lakshyajeet Dwivedee Supervisor: Dr Xavier Bresson

Travelling Salesman Problem

"Given a list of cities and the distance between each pair of cities, what is the shortest possible path that visits each city and returns to the origin"

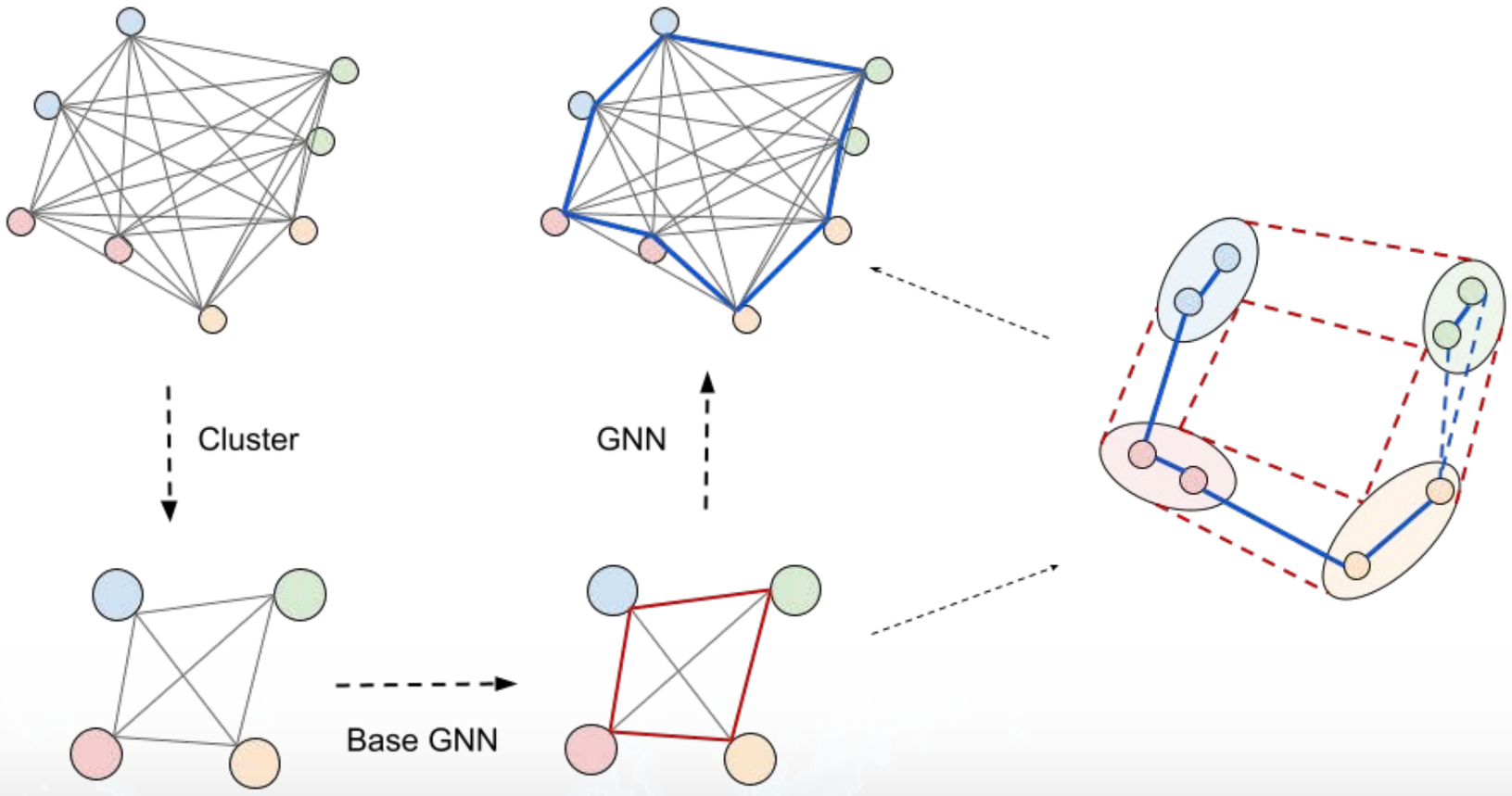
- Famous **NP-hard problem**, difficult to solve in polynomial time for large graphs.
- Relevant to fields such as supply chains, transportation, networks, biology, etc.
- Efficient solution needed for large graph sizes.

Objectives

TSP - many nodes & edges, real-time constraints

- *Exact solutions* - **Slow and infeasible** as graph size ↑
- *Heuristics* - **Require extensive research** to create
- *Graph Neural Network*s - Learn heuristics automatically, but have **inefficient O(n^2)** complexity

Solution: Using **divide-and-conquer** to make an **efficient O(n)** Graph Neural Network



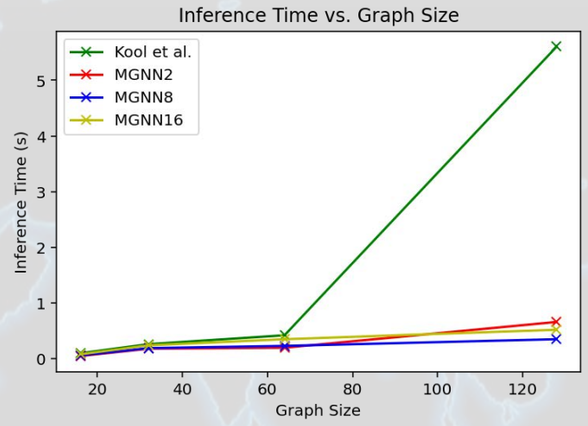
Procedure

1. Use **multilevel clustering** to decrease graph size M times
2. Use **Graph Neural Network** to solve TSP for smallest graph (base level)
3. For each child graph (increasing size):
    1. Add current node and sibling from same supernode to tour
    2. Compare sibling with just children of parent supernode's neighbor
    3. Use **Multilevel Graph Neural Network** to connect supernodes via children

*Works in **O(n) time!***

Results

- Linear runtime vs. state-of-the-art *(right)*
- Higher accuracy for lesser levels of clustering *(bottom)*



n	Optimality Gap for models (x = nodes in base level for MGNNx)			
	Kool et al.	**MGNN2**	**MGNN8**	**MGNN16**
16	0.32%(0.10s)	34.98%(0.05s)	13.83%(0.06s)	5.18%(0.08s)
32	1.03%(0.26s)	42.66%(0.18s)	39.07%(0.19s)	32.96%(0.24s)
64	4.80%(0.42s)	62.34%(0.20s)	54.24%(0.23s)	54.13%(0.35s)