

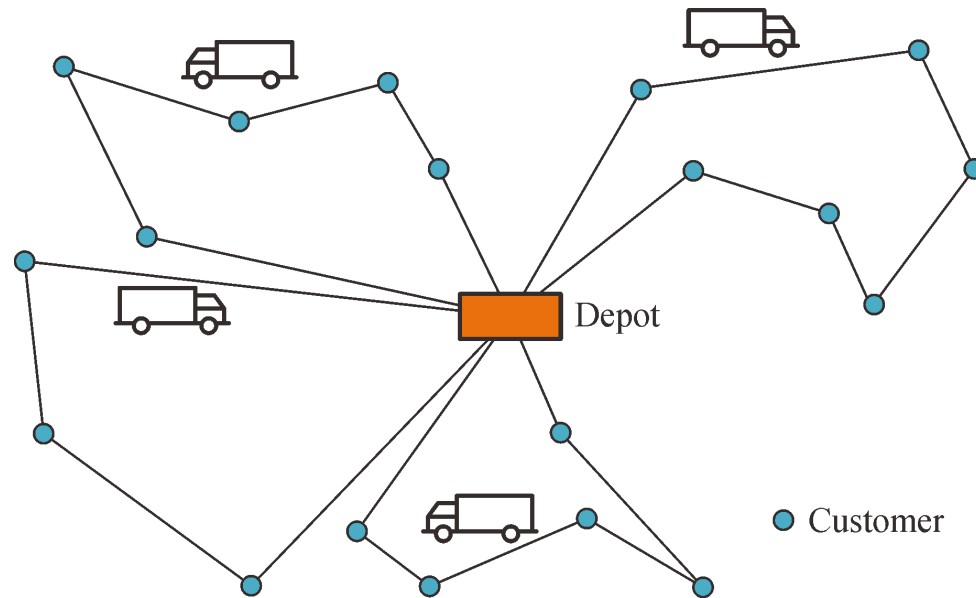


# Parallelizing VRP post-processing

Tanvir Ibna Kaisar  
ID: 4728644603  
CSCI 596 Final Project  
Instructor: Professor Aiichiro Nakano



# Background



1. NP-hard problem
2. Oftentimes post-processing needed



# Typical Post-processing

For **each tour**, we solve:

$$\begin{aligned} \min_{p, \epsilon} \quad & \sum_{i \in N} \epsilon_i^{IR} + \sum_{i \in N} \epsilon_i^S + \epsilon^{BB} \\ \text{subject to,} \quad & p_i \leq p_i^{(i,j)} + \epsilon_i^S \quad \forall i \in N_c \quad \forall j \in (N_c \cup N_r) \\ & \sum_{i \in N_c} p_i = \mathcal{C}(N_c) + \sum_{i \in N_c} \epsilon_i^{IR} + \sum_{i \in N_c} \epsilon_i^S + \epsilon^{BB} \\ & p_i \leq 2c_{i0} + \epsilon_i^{IR} \quad \forall i \in N_c \\ & p_i \geq 0 \quad \forall i \in N_c \\ & \epsilon_i^{IR}, \epsilon_i^S, \epsilon^{BB} \geq 0 \end{aligned}$$

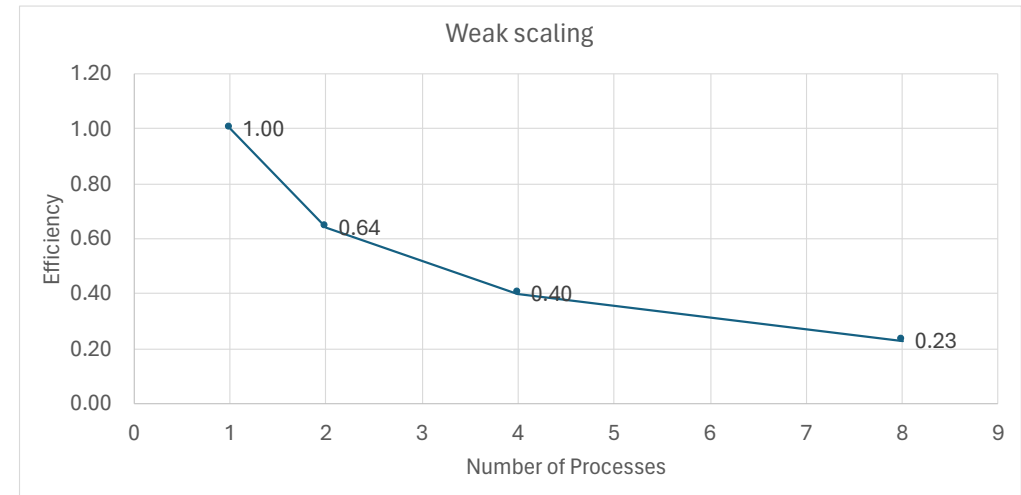
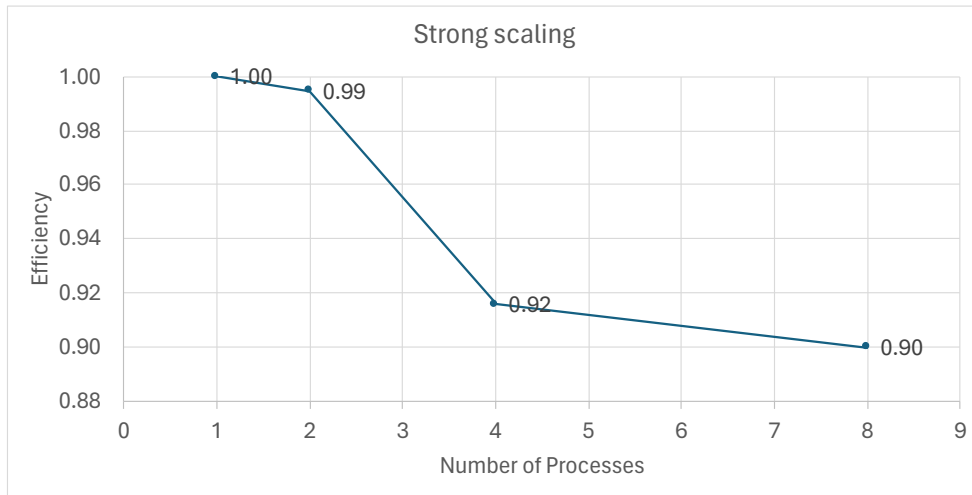
Just a flavor of how  
complicated post-  
processing could get

1. Linear program
2. Parallelization



# Result

- Typical measure of workload for LP models is the **number of nodes** (not quite, as we will see later)
- Nodes: 50, 100, 200, 400
- Number of processors: 1, 2, 4, 8





# Analysis

Why weak scaling efficiency is so bad?

num_nodes	num_var	num_constr	factor_var	factor_constr
50	162	2466	1	1
100	328	9932	2.02	4.03
200	666	39870	4.11	16.17
400	1270	149724	7.84	60.72

- Number of constraints increases at a higher rate. For example, 400 node problem has 60 times more constraints than 50 node problem.
- We cannot know apriori how the number constraints will grow for linear programs.
- Many constraints are redundant (for example  $2 < 10$  and  $5 < 10$ , only the first one suffices, but again we do not know this apriori).