

关于多旅行商问题(MTSP)的 启发式算法研究

——《智能优化算法》结课汇报



小组成员：赵心怡，卢星汝，林子业，田斐然，陈凯强

2024.05.08

目录



研究背景



问题界定



建立模型



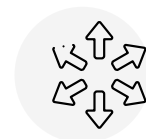
数据来源与Gurobi最优解



GA、SA求解与模型对比



改进点



实际应用

A group of people in business attire are leaning over a large, light-colored table, engaged in a collaborative meeting or discussion. Their hands are visible on the table, some pointing at documents or objects. The background is bright and out of focus, suggesting a modern office or meeting room.

Part 1

研究背景与数学模型

研究背景：相比经典TSP，现实中更多场景涉及多任务分配与优化，对应MTSP问题

- 经典TSP 问题是一种最基础的求最短路径规划问题，它只需要考虑路程最短这一单一条件约束，而生活中大部分实际应用问题却不能被归纳为TSP问题，例如**物流配送、人员调度、巡查救援**等一些涉及到多任务分配与优化的场景
- 多旅行商涉及到的内容更复杂，研究成果也相对较少。目前很多研究只停留在优化数学模型的阶段，而没有考虑特定应用领域

应用领域	问题	解决模型和方法举例
 物流运输与配送	<ul style="list-style-type: none">• 多个快递员的任务分配和路线优化的方法	<ul style="list-style-type: none">• 任务均衡的MTSP（用遗传算法求解；用仿生算法优化 [袁雨果等]
 无人机运输小包裹、 移动机器人分拣快递货物	<ul style="list-style-type: none">• 仓储物流过程中订单拣选机器人实时路径规划问题• “最后一英里”的物流交付系统	<ul style="list-style-type: none">• 多旅行商，通过结合改进的自适应遗传算法实现批量选择路径规划 [潘成浩等]• 一辆送货卡车与一组无人机进行协同工作，相互配合，使距离仓库较远的客户能够接收无人机的交付 [Murray等]
 应急救援	<ul style="list-style-type: none">• 应急物资分配、无人机调度优化和路径优化等	<ul style="list-style-type: none">• 多机器人搜索和救援任务分配的组合优化数学模型。设计的算法可以满足在机器人损毁、机器人数量新增或任务新增等动态环境变化的情况下快速适应动态环境的要求 [陈昕叶等]
 军事	<ul style="list-style-type: none">• 如何在攻击地面多目标群的过程中找到最优的攻击路径	<ul style="list-style-type: none">• 将攻击地面多目标群过程中的路径规划问题转化为 MTSP 问题。并提出了一种基于MTSP-GA的模型计算方法，分别得到了单基地起飞和双基地起飞的最优路径规划结果 [徐国训等]

MTSP相比TSP目标和约束的种类大大增加，可以根据终点和仓库数量分为4类问题

经典TSP

问题

- 旅行商问题：有1个推销员与K个城市的路径规划，确定这个推销员的路线，有序地通过它们。

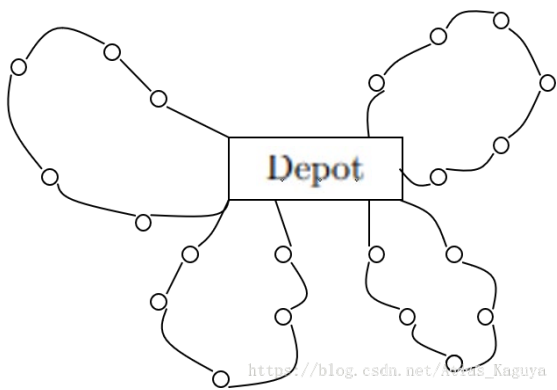
目标

- 最小化路线总距离 / 时间

约束

- 终点：最终需要回到起点

图示



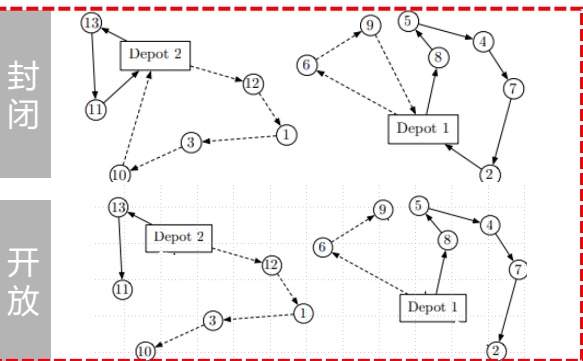
经典MTSP（蓝色）及其变体

- 多旅行商问题：确定一组推销员的路线，所有销售人员都从家乡（仓库）出发并经过一系列城市，每个城市只访问一次

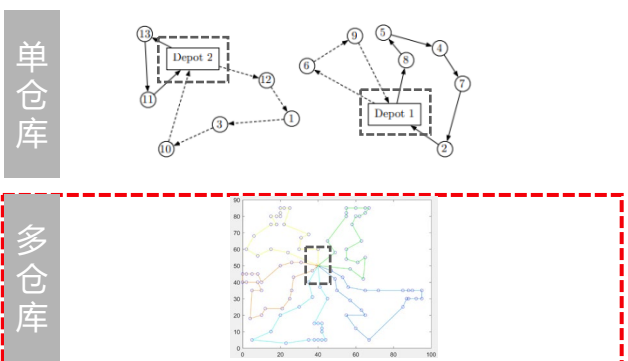
- 最小化总距离 / 时间 / 能源消耗 / 旅行商数目
- 实际应用中，可能既要考虑总路程最小化，又要考虑各推销员之间的工作量的平衡；既要考虑提高公司的收益，又要考虑每个客户服务的等待时间问题。

- 仓库：分单仓库和多仓库、固定仓库和移动仓库
- 终点：在经典的MTSP中，旅行商的路径是封闭的（回到各自起点），但在某些实际应用中，旅行商可以不需要返回仓库，停留在最后访问的城市即可（开放路线）。
- 其他：能量约束、容量约束（此时同MDVRP）、时间窗约束等

分类维度1：终点是否为原点



分类维度2：仓库的数量



注：代表后续我们的模型的假设：目标为最小化总距离，多个depot且不固定（自动选择最优），路线分封闭和开放两种情形；经典MTSP及其变体中蓝色代表经典MTSP的假设

参考：张硕航,郭改枝.多旅行商模型及其应用研究综述[J].计算机科学与探索,2022,16(07):1516-1528.

根据4类MTSP情形，对应4种目标函数模型

- 设定n个目标的集合为 $\{T, T_{i_1}, T_{i_2}, \dots, T_{i_n}\}$ ，m个机器人的集合为 $\{R_1, R_2, \dots, R_m\}$

单仓库

封闭路线

- m个推销员从**同一起点**出发，最终**返回起点**

$$C(Tour_{R_i}) = C(D, T_{i_1}) + \sum_{k=1}^{r-1} C(T_{i_k}, T_{i_{k+1}}) + C(T_{i_r}, D)$$

开放路线

- 旅行商在最后一处访问城市任务终止，**不再回到起点**

$$C(Tour_{R_i}) = C(D, T_{i_1}) + \sum_{k=1}^{r-1} C(T_{i_k}, T_{i_{k+1}})$$

多仓库

- m个推销员从**不同起点**出发，遍历了目标城市后最终**返回到各自的起点**

$$C(Tour_{R_i}) = C(D_i, T_{i_1}) + \sum_{k=1}^{r-1} C(T_{i_k}, T_{i_{k+1}}) + C(T_{i_r}, D_i)$$

- m个旅行商从**不同起点**出发，任务结束后停留在当前城市，无须返回各自起点

$$C(Tour_{R_i}) = C(D_i, T_{i_1}) + \sum_{k=1}^{r-1} C(T_{i_k}, T_{i_{k+1}})$$

我们以多仓库、封闭路线情形为例，建立具体的数学模型，用于后续Gurobi求解

数学模型

解释

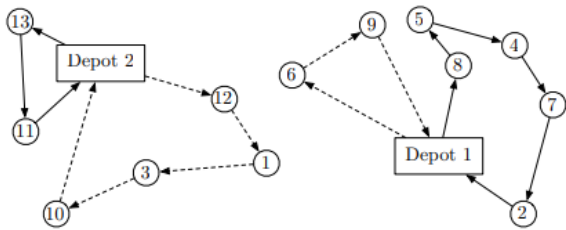
变量和
目标

$$x_{ij}^k = \begin{cases} 1, & \text{如果销售人员} k \text{在访问} i \text{之后紧接着访问} j \\ 0, & \text{otherwise} \end{cases}$$

$$z_i = \begin{cases} 1, & \text{节点} i \text{是任一销售人员的起始节点} \\ 0, & \text{otherwise} \end{cases}$$

u_i = 结点 i 的访问等级，即第几个被访问

$$\text{Minimize } \sum_{i=1}^n \sum_{j=1; i \neq j}^n (d_{ij} \sum_{k=1}^m x_{ij}^k) \quad (1)$$



—— 目标：最小化 m 次旅行路径长

约束

Subject to:

$$\sum_{i=1; i \neq j}^n \sum_{k=1}^m x_{ij}^k = 1 \quad \forall j = 1, 2, \dots, n \quad (2)$$

$$\sum_{i=1; i \neq p}^n x_{ip}^k - \sum_{j=1; j \neq p}^n x_{pj}^k = 0 \quad \forall p = 1, 2, \dots, n, k = 1, 2, \dots, m \quad (3)$$

$$\sum_{i=1}^n \sum_{j=1; i \neq j}^n x_{ij}^k \geq 1 \quad \forall k = 1, 2, \dots, m \quad (4)$$

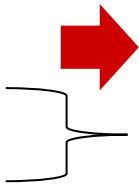
$$u_i - u_j + L \sum_{k=1}^m x_{ij}^k \leq L - 1 + Lz_j \quad \forall i, j = 1, 2, \dots, n; i \neq j \quad (5)$$

$$1 \leq u_i \leq L \quad \forall i = 1, 2, \dots, n \quad (6)$$

$$\sum_{i=1}^n z_i = m \quad (7)$$

$$x_{ij}^k \in (0, 1) \quad \forall i, j = 1, 2, \dots, n; i \neq j; k = 1, 2, \dots, m \quad (8)$$

$$z_i \in (0, 1) \quad \forall i = 1, 2, \dots, n \quad (9)$$



—— 确保每个结点只被访问一次

—— 流量守恒约束集，如果访问某个节点，最终必须离开该节点

—— 确保使用每个销售人员

—— 消除子环路， L 表示任何销售人员可以访问的最大节点数。由于每个销售人员在他/她的行程中至少应该访问两个节点， $L = n - 2m + 2$ 。



Part 2

算法设计与实验结果

算法流程和总览：运用Gurobi、GA和SA三种算法对MTSP问题求解并进行了对比

Gurobi

建立数学模型

在访问*i*之后紧接着访问*j*
otherwise
销售人员的起始节点
otherwise

$u_i =$ 结点*i*的访问等级，即第几个被访问

Minimize $\sum_{i=1}^n \sum_{j=1, j \neq i}^n (d_{ij} \sum_{k=1}^m x_{ij}^k)$ (1)

Subject to:

$\sum_{i=1, i \neq j}^n \sum_{k=1}^m x_{ij}^k = 1 \quad \forall j = 1, 2, \dots, n$ (2)

$\sum_{i=1, i \neq p}^n x_{ip}^k - \sum_{j=1, j \neq p}^n x_{pj}^k = 0 \quad \forall p = 1, 2, \dots, n, k = 1, 2, \dots, m$ (3)

$\sum_{i=1}^n \sum_{j=1, j \neq i}^m x_{ij}^k \geq 1 \quad \forall k = 1, 2, \dots, m$ (4)

$u_i - u_j + L \sum_{k=1}^m x_{ij}^k \leq L - 1 + Lz_{ij} \quad \forall i, j = 1, 2, \dots, n; i \neq j$ (5)

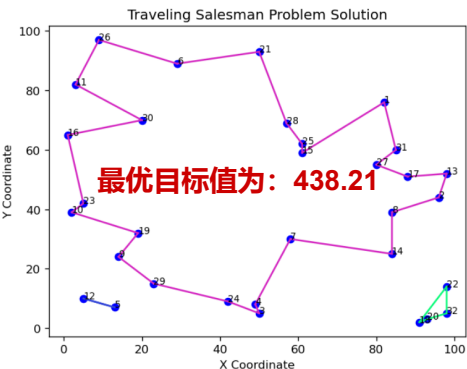
$z_{ij} \in \{0, 1\}$ (6)

$z_{ij} \in \{0, 1\}$ (7)

$z_{ij} \in \{0, 1\}$ (8)

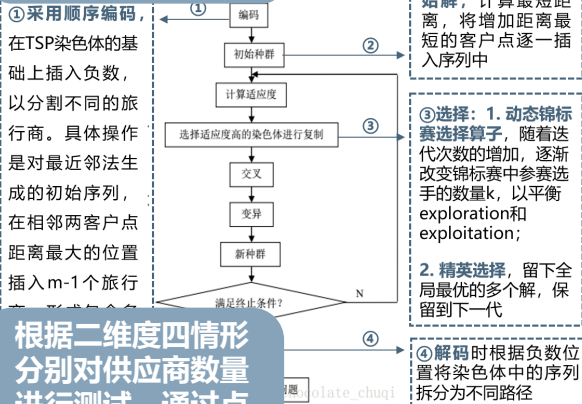
$z_{ij} \in \{0, 1\}$ (9)

Gurobi求解

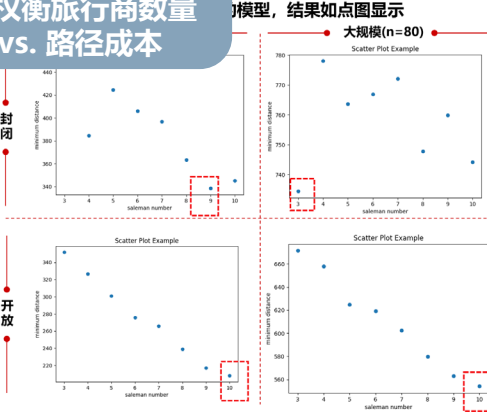


GA

设计编码、初始解、选择、解码过程

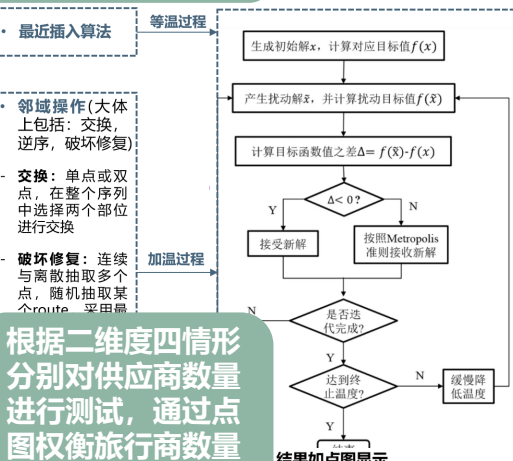


根据二维度四情形分别对供应商数量进行测试，通过点图权衡旅行商数量 vs. 路径成本

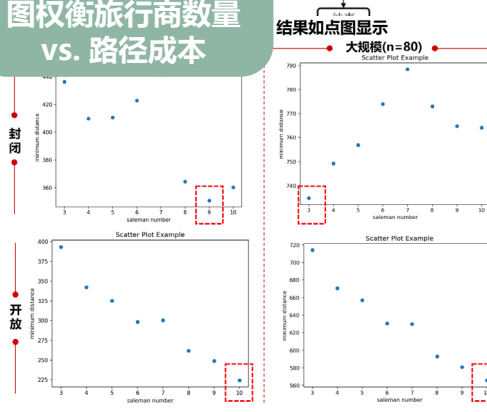


SA

设计初始解、邻域操作和禁忌部分



根据二维度四情形分别对供应商数量进行测试，通过点图权衡旅行商数量 vs. 路径成本



结果对比

以32个供应商封闭情形为例

	Gurobi	GA	SA
求解时间	~1min	设定1min/次 (8次取最小)	设定1min/次 (8次取最小)
最优解	438.21	456.02	438.35

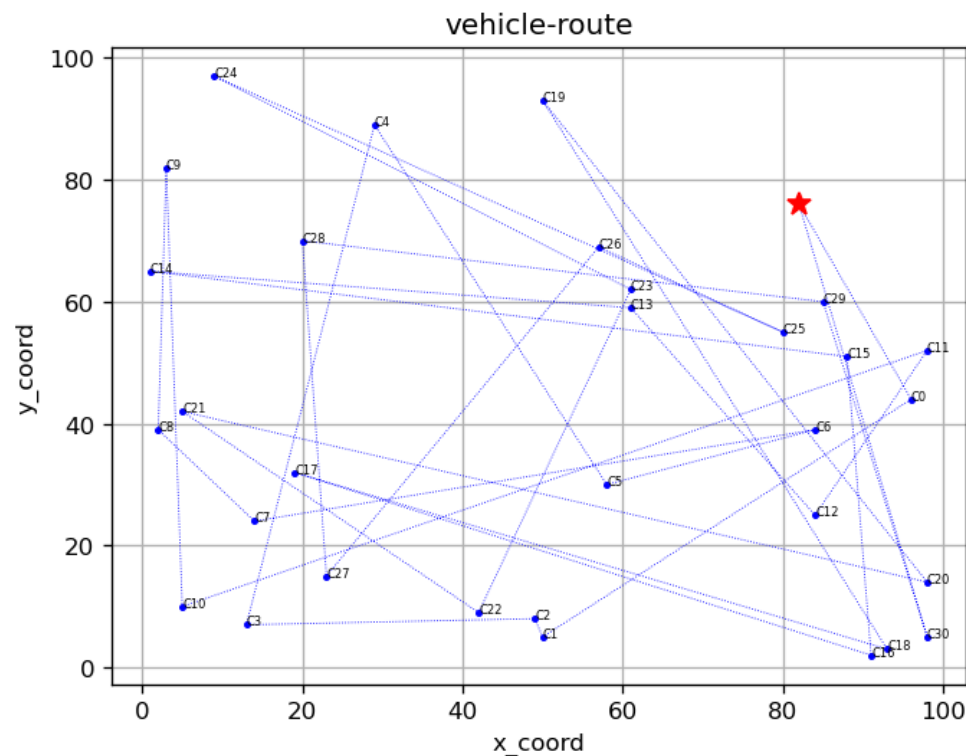
大规模问题，同样对三个算法进行了比较，并对算法优化程度进行评估

	Gurobi	GA	SA
小规模 (N=32)	最优解: 438.21	优化前: 499.85 优化后: 456.02	优化前: 1331 优化后: 438.35
大规模 (N=80)	无法实现	优化前: 1718.94 优化后: 1043.39	优化前: 3056 优化后: 787.47

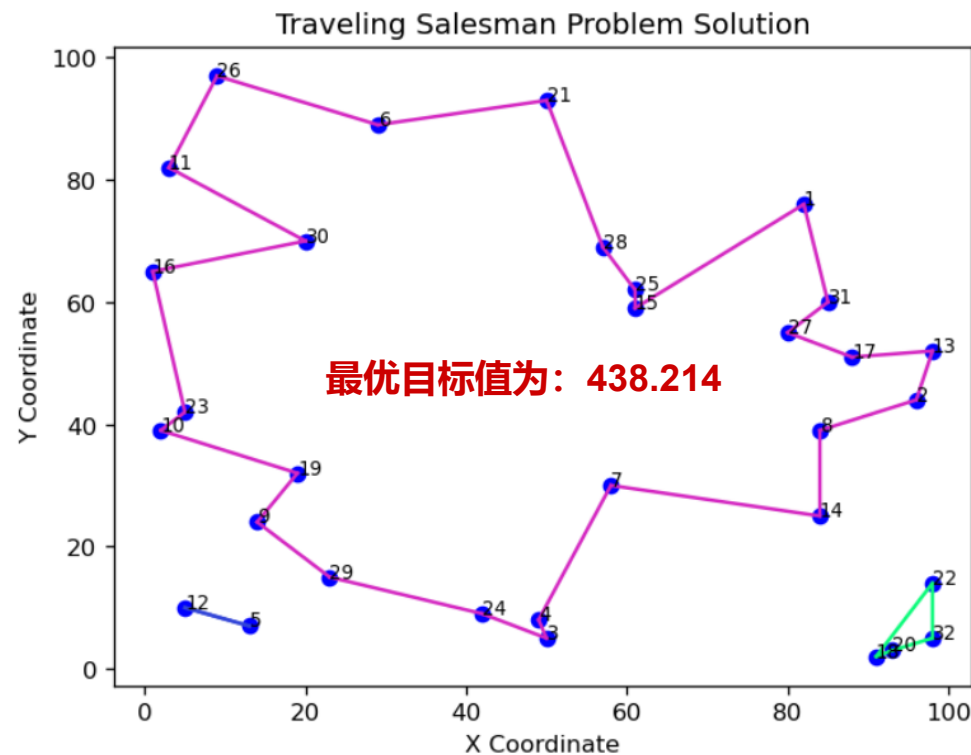
数据来源： CVRPLIB 的 A-n32-k5 标准数据集；用Gurobi求解出最优路径和目标值

- CVRPLIB(Capacitated Vehicle Routing Problem Library) 是提供车辆路径问题数据集的平台。它的测试数据为了方便不同平台，不同代码的测试和迁移，针对CVRP问题具有标准的数据格式——**VRPLIB**格式，是一种存放CVRP问题数据的经典形式。
- **VRPLIB** 是一个专门用于处理带容量车辆路径问题 (CVRP) 实例的 Python 包。它最主要的特点是可以解析和编写VRPLIB标准格式的数据，以及从CVRPLIB上下载标准测试数据。依赖于numpy库。我们取**A-n32-k5**这一标准数据集，并根据前面的数学模型用Gurobi建模求解。（尝试更大规模，如n=80时Gurobi无法求解）

数据分布



Gurobi求解出的最优路径和目标值

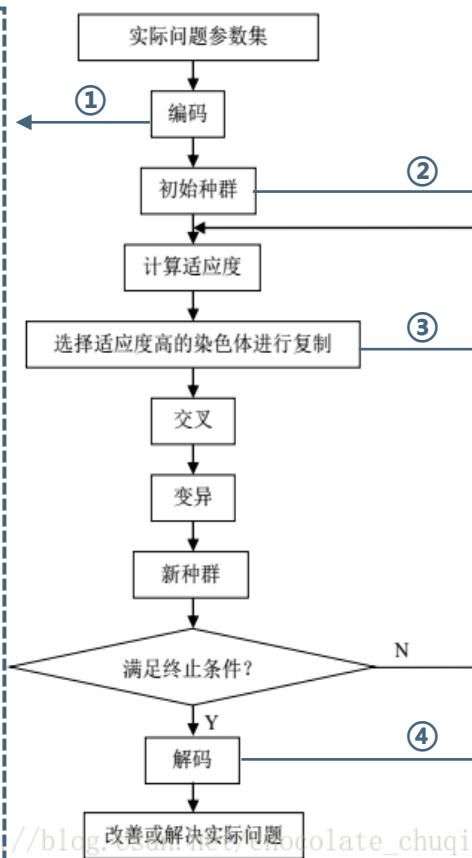


GA算法：优化点主要在于编码、初始解、选择过程和解码过程

算法设计

- 我们对编码、初始解、选择过程和解码过程进行了设计

①采用顺序编码，在TSP染色体的基础上插入负数，以分割不同的旅行商。具体操作是对最近邻法生成的初始序列，在相邻两客户点距离最大的位置插入m-1个旅行商，形成包含多旅行商的初始解。



②最近邻法生成初始解，计算最短距离，将增加距离最短的客户点逐一插入序列中

③选择：1. 动态锦标赛选择算子，随着迭代次数的增加，逐渐改变锦标赛中参赛选手的数量k，以平衡 exploration和 exploitation；

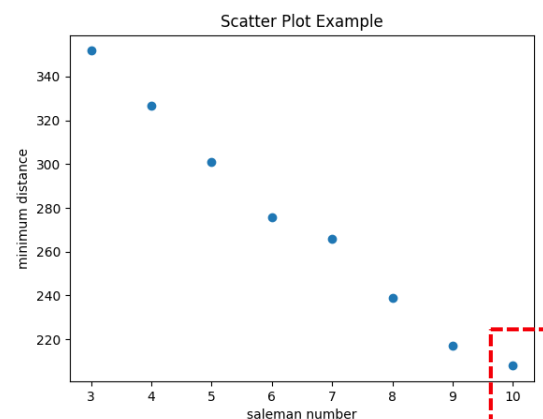
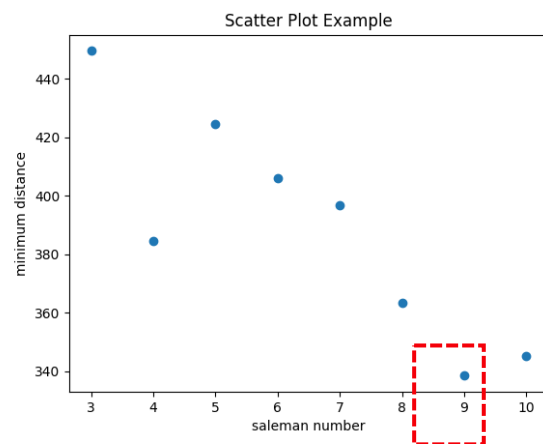
2. 精英选择，留下全局最优的多个解，保留到下一代

④解码时根据负数位置将染色体中的序列拆分为不同路径

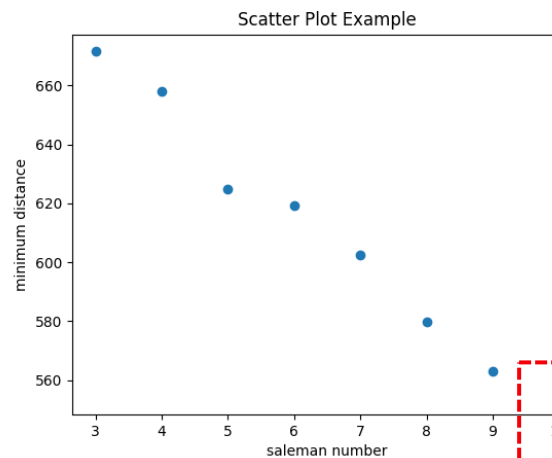
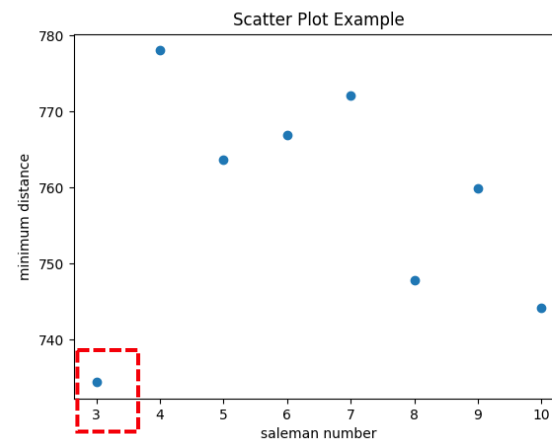
结果

- 输出旅行商数量为3~10的点图便于权衡路径成本和旅行商成本

小规模(n=32)



大规模(n=80)

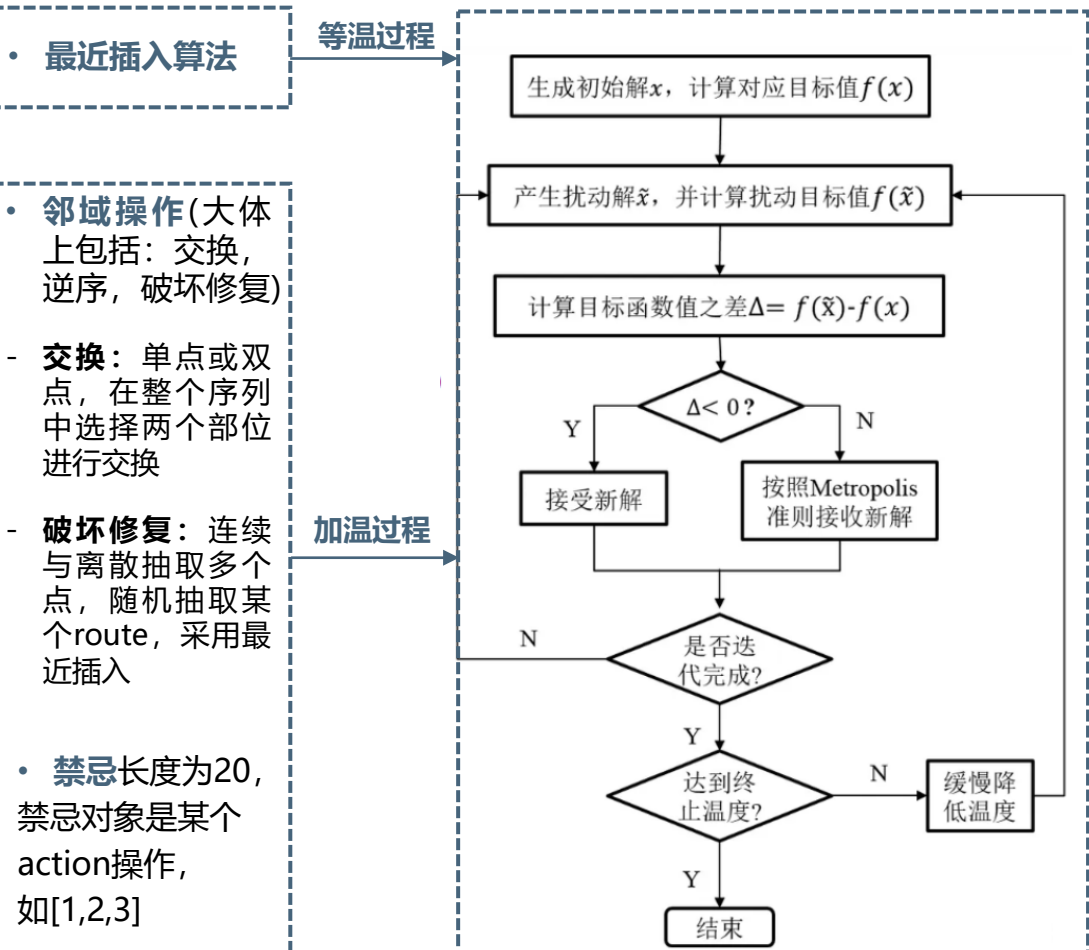


注：结果均接受单次解（即一个旅行商可以只经过一个城市）

SA算法：优化点主要在初始解、邻域操作和禁忌部分

算法设计

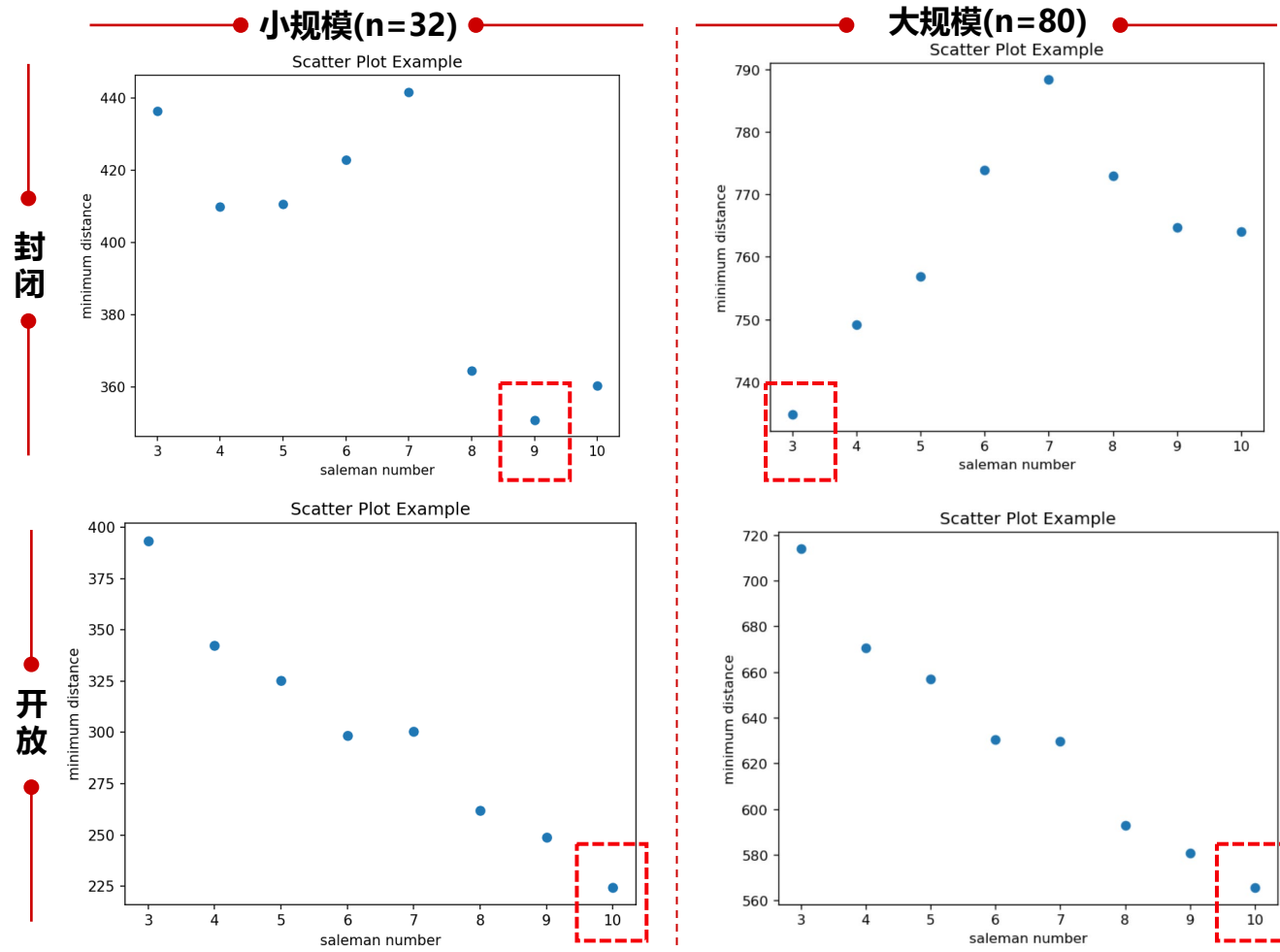
SA算法主要对初始解、邻域操作和禁忌部分进行了设计



注：结果均接受单次解（即一个旅行商可以只经过一个城市）

结果

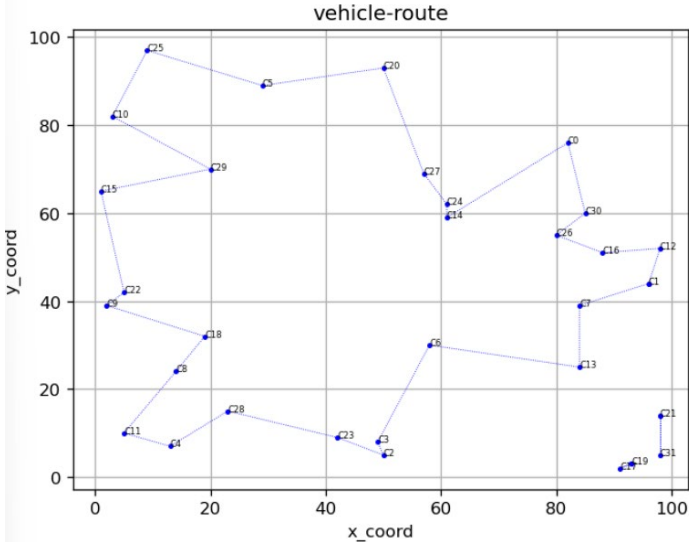
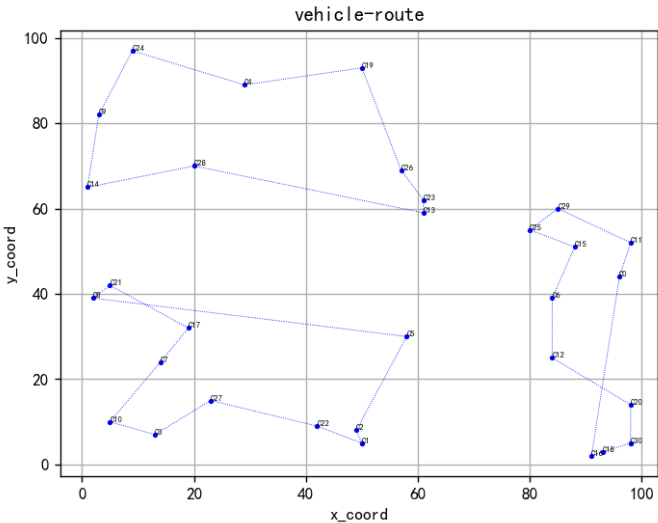
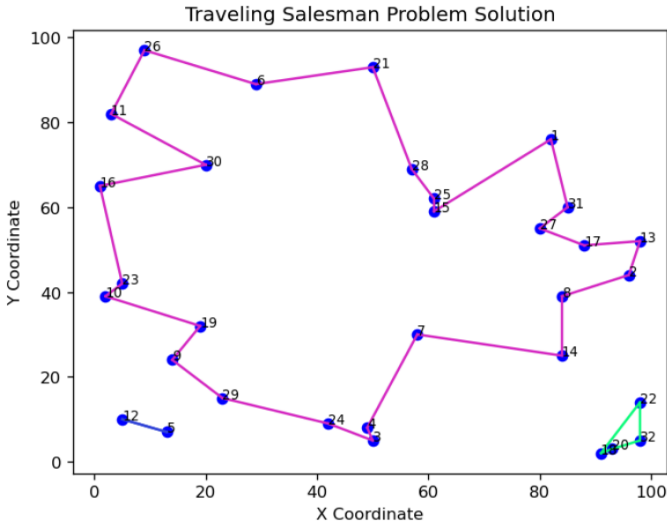
依次执行旅行商数量为3~10的模型，结果如点图显示



结果：小规模(32个供应商)封闭情形，GA&SA均接近Gurobi最优解

	Gurobi	GA	SA
求解时间	~1min	设定1min/次（8次取最小）	设定1min/次（8次取最小）
最优解	438.21	456.02	438.35

路径



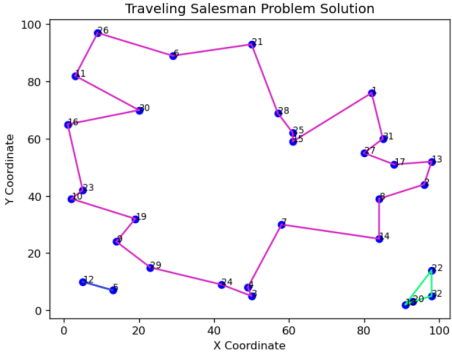
注：SA和GA均采用每次1分钟的时间窗（与Gurobi近似），均运行8次取目标函数最小的情况；均不接受单次解

推广：小规模问题GA/SA均接近最优解，大规模问题SA效果更佳；算法优化效果显著

小规模
(N¹=32)

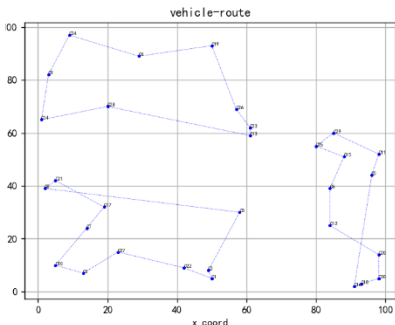
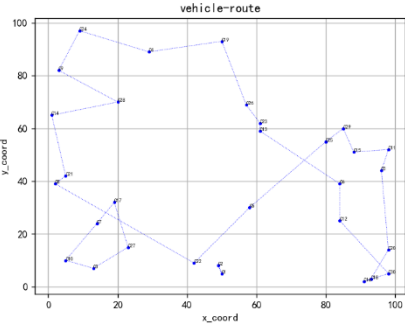
Gurobi

最优解: 438.21



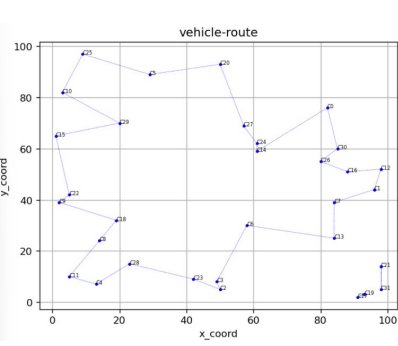
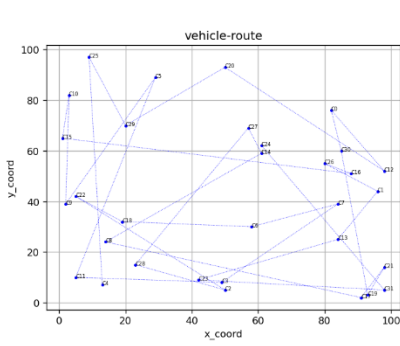
GA

优化前: 499.85 优化后: 456.02



SA

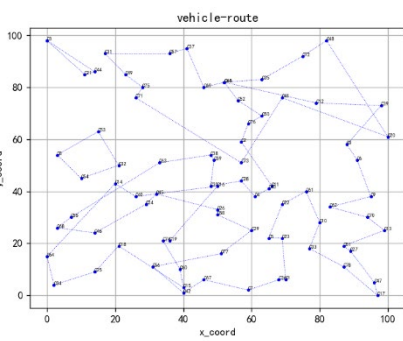
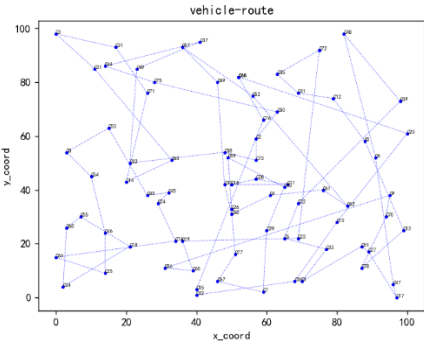
优化前: 1331 优化后: 438.35



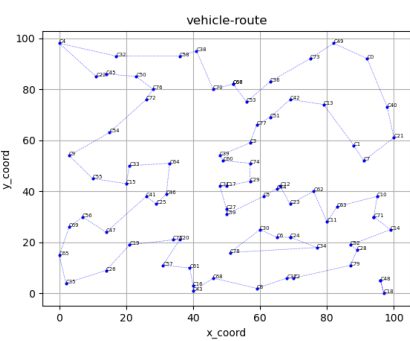
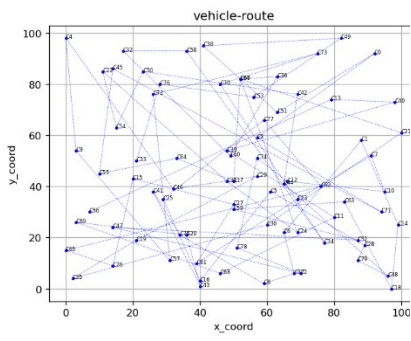
大规模
(N=80)

无法实现

优化前: 1718.94 优化后: 1043.39



优化前: 3056 优化后: 787.47



注：1. N代表供应商数量；均不接受单次解



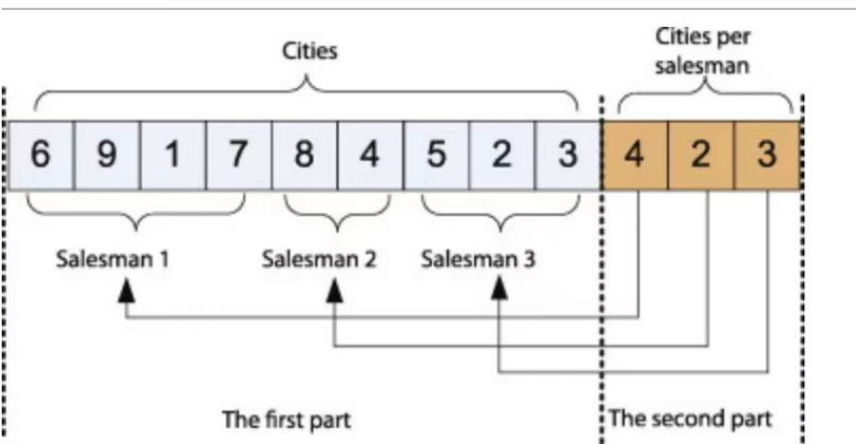
Part 3

展望

进一步完善方向：尝试其他算法和算法组合；结合具体商业问题求解

• 参考论文，对算子进行创新：

- **编码：**将染色体分成两部分。长度的第一部分代表n个城市的排列，长度的第二部分给出分配给每个推销员的城市数量。因此，在此表示中染色体的总长度为n+m。染色体第二部分中存在的m值之和必须等于n才能表示有效的解决方案



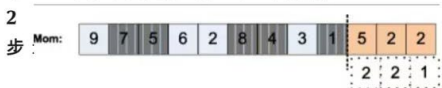
- **新的交叉方法：**进行交叉的时候，每个旅行商访问的城市数量也会改变，但传统的交叉方法就不会改变这个部分，只会改变前面的城市顺序

步 将一对染色体初始化为父母

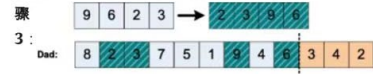


下载：下载全尺寸图像

第 为每个推销员随机选择一个基因片段



步 根据爸爸染色体的第一部分改组基因位置

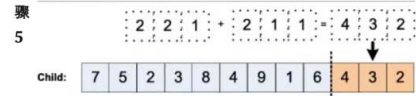


步 为每个推销员添加基因

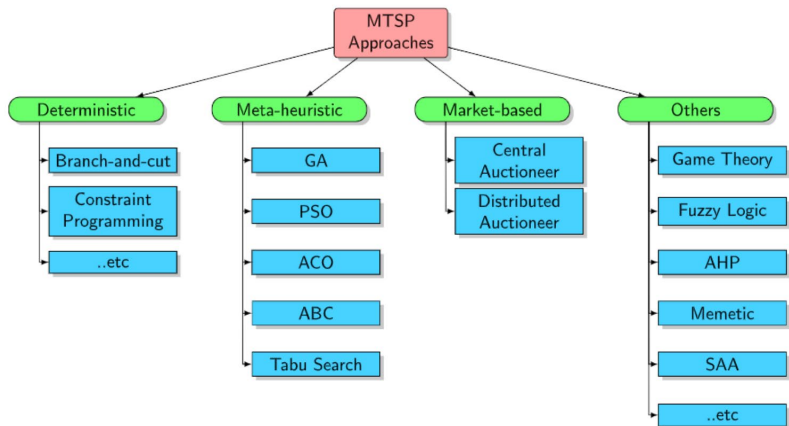


下载：下载全尺寸图像

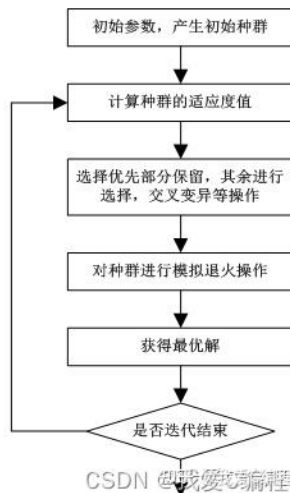
步 构建孩子的两部分染色体



• 尝试其他算法

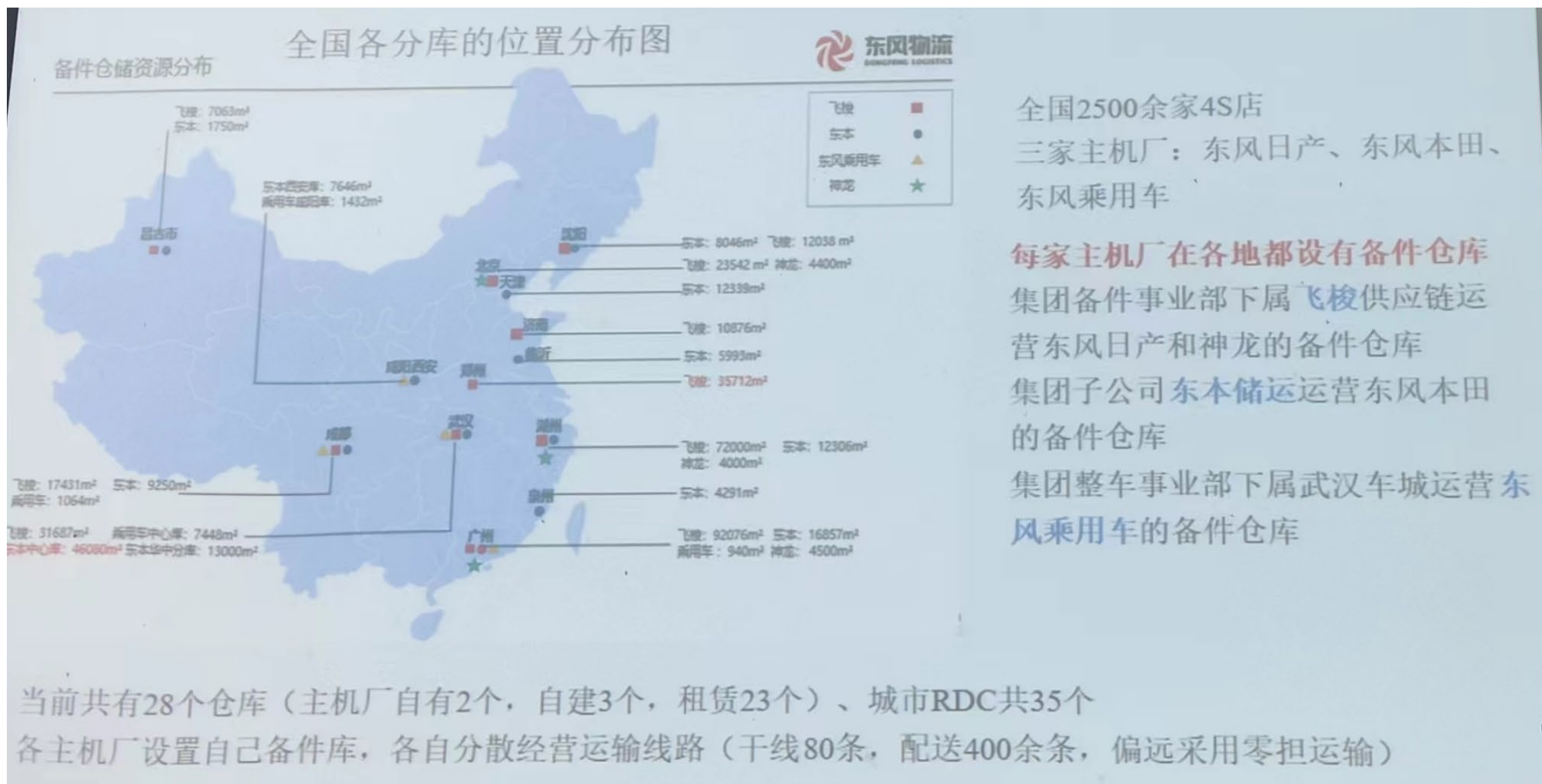


• 尝试算法嵌套（如GA+SA）




尝试与商业结合，解决选址、配送路径规划等实际商业问题 (1/2)

- 借鉴许明辉老师CVRP+设施选址相关项目



尝试与商业结合，解决选址、配送路径规划等实际商业问题 (2/2)

- 借鉴许明辉老师CVRP+设施选址相关项目

模型构建思路	以东北地区为例
<ul style="list-style-type: none">分库1个（沈阳分库），RDC有3个（哈尔滨，长春，大连）4S店有133个将133个4S店进行编号 ($j \in J = \{1, 2, \dots, 133\} = J_1 \cup J_2$)。$J_1$：所有采用干线、支线、专线进行配送的4S店的集合J_2：其他采用零担方式配送的4S店集合	<p>东北地区4S店分布</p> 
<p>决策：</p> <ul style="list-style-type: none">分库位置（只选1个分库）RDC的数量及相应位置确定好以上决策后的配送路径规划	

附：主要参考文献

- [1] Shuai Yuan, Bradley Skinner, Shoudong Huang, Dikai Liu, A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms, European Journal of Operational Research, Volume 228, Issue 1, 2013, Pages 72-82, ISSN 0377-2217, <https://doi.org/10.1016/j.ejor.2013.01.043>.
- [2] Korhan Karabulut, Hande Öztop, Levent Kandiller, M. Fatih Tasgetiren, Modeling and optimization of multiple traveling salesmen problems: An evolution strategy approach, Computers & Operations Research, Volume 129, 2021, 105192, ISSN 0305-0548, <https://doi.org/10.1016/j.cor.2020.105192>.
- [3] 张硕航,郭改枝.多旅行商模型及其应用研究综述[J].计算机科学与探索,2022,16(07):1516-1528.
- [4] 孙鉴,刘品,李昊,等.基于Spark的双阶段SA及GA求解MTSP[J/OL].郑州大学学报(工学版):1-9[2024-05-07].<https://doi.org/10.13705/j.issn.1671-6833.2024.01.019>.

感谢倾听!

关于**多旅行商问题(MTSP)**的
启发式算法研究



小组成员：赵心怡，卢星汝，林子业，田斐然，陈凯强

2024.05.08