

TSP

王英梟

211840213

wangbottlecap@gmail.com

April 17, 2025

Abstract

离散优化课程作业，主要包括求解器 Gurobi、COPT 对旅行商问题样例 d657, rat575, u1060 的实现。对两种求解器的调用均放弃 MTZ 子环约束，转而使用 Callback Module 进行 Lazy Constraint 限制。此外，添加了最近邻算法以加速寻找初始解，并在适当的地方进行求解器参数调优。

Contents

1 数学模型	1
2 Gurobi	1
3 COPT	2
4 代码公开	2

1 数学模型

定义两个城市 i, j 间的距离为其二维坐标下的欧几里得距离，目标函数因此被定义为路径 ω 在此城市距离定义下的总长度。物理约束仅添加“每个城市 i 的度 (degree) 为 2”，子环在 Callback 中考虑。

2 Gurobi

使用 Gurobi 进行计算的配置如下：

1. Callback: 只有找到合法子环，才添加 lazy constraint 把该子环排除；放弃在硬约束中排除所有子环

2. 求最短子环的函数不会找到对称子环，以破解对称性
3. 在 Gurobi 求解之前使用最近邻算法生成初始可行路径，并设置为初始解
4. 参数调优：

- `model.setParam("MIPFocus", 1)`
- `model.setParam("ImproveStartTime", 5)`
- `model.setParam("ImproveStartNodes", 100)`
- `model.setParam("Cuts", 1)`
- `model.setParam("Heuristics", 0.8)`
- `model.setParam("Presolve", 1)`
- `model.setParam("Threads", 8)`

用于数值比较的参数配置（如计算时间 `=timeLimit`，绝对最优间隙 `=0.000001` 等）已略去。

3 COPT

使用 COPT 进行计算的配置如下：

1. Callback: 只有找到合法子环，才添加 lazy constraint 把该子环排除；放弃在硬约束中排除所有子环
2. 求最短子环的函数不会找到对称子环，以破解对称性
3. 在 COPT 求解之前使用最近邻算法生成初始可行路径，并设置为初始解
4. 参数调优：

- `model.setParam(COPT.Param.HeurLevel, 3)`
- `model.setParam(COPT.Param.Threads, 8)`
- `model.setParam(COPT.Param.Presolve, 2)`

用于数值比较的参数配置（如计算时间 `=timeLimit`，绝对最优间隙 `=0.000001` 等）已略去。

4 代码公开

代码已在 Github 公开，具体细节参见<https://github.com/Bottlecapless/TSP.git>，谢谢！