Solving the Green Vehicle Routing Problem

Juho Andelmin          Enrico Bartolini¹


• Andelmin, J., Bartolini, E. A Multi-Start Local Search Heuristic for the Green Vehicle Routing Problem Based on a Multigraph Reformulation. Submitted to Computers and Operations Research

¹ RWTH Aachen University
School of Business and Economics
A fleet of vehicles based at a **depot** is to serve a set of customers
- Customers have known service times
- Vehicles have limited fuel capacity
- Vehicles can visit refueling stations to refuel

**Objective:** Design a set of **vehicle routes** so that
- Every customer is served
- Duration of each route \( \leq T \)
- Sum of route costs is minimized
Simple example: 9 customers, electric vehicles

- Vehicle speed: 90 km/h
- Service time: 5 min
- Charging delay: 20 min
- Max route duration: 12 h
Optimal solution with driving range = ∞

Optimal cost
694.71 km

- Vehicle speed: 90 km/h
- Service time: 5 min
- Charging delay: 20 min
- Max route duration: 12 h
Optimal solution with driving range = 200 km

Optimal cost
823.26 km

- Vehicle speed: 90 km/h
- Service time: 5 min
- Charging delay: 20 min
- Max route duration: 12 h
Optimal solution with driving range = 160 km

Optimal cost
1148.08km

- Vehicle speed: 90 km/h
- Service time: 5 min
- Charging delay: 20 min
- Max route duration: 12 h
Refuel paths

- **Refuel path**: a simple path between two customers that visits a subset of refueling stations

![Diagram of refuel paths]

- Many refuel paths are dominated
- Example:
  - **Green path** \( i \rightarrow c \rightarrow j \) is dominated by
  - **orange one** \( i \rightarrow b \rightarrow j \)
We model the G-VRP on a multigraph $\mathcal{G}$ with one arc for each non-dominated refuel path.
Multi-Start Local Search Heuristic (MSLS)

Three phases

1) Iteratively construct new solutions
2) Store vehicle routes forming these solutions in a pool $\mathcal{R}$
3) Find a set of routes in $\mathcal{R}$ that gives least cost solution

Example operators used in phase 1

- Clarke and Wright Merge
- Customer relocate

[Diagram of Clarke and Wright Merge]

[Diagram of Customer relocate]
Exact algorithm

- Set partitioning formulation (SP)
  - Each possible vehicle route serves a subset of customers
  - Find least cost set of routes serving each customer exactly once

\[
\begin{align*}
\text{(SP)} & \quad \min \sum_{l \in \mathcal{R}} c_l x_l \\
\text{s.t.} & \quad \sum_{l \in \mathcal{R}} a_{ii} x_l = 1 \quad \forall i \in N \\
& \quad x_l \in \{0,1\} \quad \forall l \in \mathcal{R}
\end{align*}
\]

\[c_l:\text{ cost of route } l\]
\[x_l:\text{ 0-1 variable equal to 1 if route } l \text{ is in solution}\]
\[a_{ii}:\text{ 0-1 coefficient equal to 1 if route } l \text{ serves customer } i\]
\[\mathcal{R}:\text{ index set of all possible vehicle routes}\]
\[N: \text{ set of customers}\]

**Phase 1:**
- Compute lower bound LB by solving Linear Programming relaxation of SP with Subset Row [4], Weak Subset Row [1], and k-path cuts [6]
- Compute upper bound UB with the MSLS heuristic

**Phase 2:**
- Enumerate all routes \(\mathcal{R}^*\) having reduced cost \(\leq UB - LB\)
- Solve SP using only the routes in \(\mathcal{R}^*\) \(\rightarrow\) optimal solution
- If all routes \(\mathcal{R}^*\) cannot be enumerated optimality not guaranteed
Benchmark problems:
- 56 instances with 20-500 customers and 3-28 stations

Heuristic: best new solutions to instances with 111-500 customers
- Compared to 7 state-of-the-art heuristics [2][3][5][7][8][9]

Exact algorithm:
- Instances up to 111 customers 28 stations solved to optimality
- Best exact from literature [5] solves up to 20 customer instances

**Computational results**

**Instance name example:**
75c_21s: 75 customers 21 stations

<table>
<thead>
<tr>
<th>Inst.</th>
<th>Opt</th>
<th>%LB</th>
<th>Time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>75c_21s</td>
<td>*</td>
<td>0.00</td>
<td>5217</td>
</tr>
<tr>
<td>75c_28s</td>
<td>*</td>
<td>0.00</td>
<td>5582</td>
</tr>
<tr>
<td>100c_21s</td>
<td>*</td>
<td>0.00</td>
<td>5206</td>
</tr>
<tr>
<td>100c_28s</td>
<td>*</td>
<td>0.00</td>
<td>6531</td>
</tr>
<tr>
<td>111c_28s</td>
<td>*</td>
<td>0.13</td>
<td>11265</td>
</tr>
<tr>
<td>200c_21s</td>
<td></td>
<td>0.60</td>
<td>28135</td>
</tr>
<tr>
<td>250c_21s</td>
<td></td>
<td>0.57</td>
<td>27939</td>
</tr>
<tr>
<td>300c_21s</td>
<td></td>
<td>0.84</td>
<td>25903</td>
</tr>
</tbody>
</table>

%LB = \left( \frac{UB - LB}{UB} \right) \times 100\%
Optimal solution to 111c_28s
Optimal solution to Distance-constrained CVRP instance CMT6

<table>
<thead>
<tr>
<th>Instance</th>
<th>n</th>
<th>s</th>
<th>UB</th>
<th>Opt</th>
<th>%LB</th>
<th>%LB+</th>
<th>Time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMT6</td>
<td>50</td>
<td>0</td>
<td>555.43</td>
<td>*</td>
<td>3.17</td>
<td>0.00</td>
<td>573</td>
</tr>
</tbody>
</table>
Optimal solution to Distance-constrained CVRP instance CMT7

<table>
<thead>
<tr>
<th>Instance</th>
<th>( n )</th>
<th>( s )</th>
<th>( UB )</th>
<th>Opt</th>
<th>%LB</th>
<th>%LB⁺</th>
<th>Time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMT7</td>
<td>75</td>
<td>0</td>
<td>909.68</td>
<td>*</td>
<td>1.65</td>
<td>0.15</td>
<td>1290</td>
</tr>
</tbody>
</table>
Heuristic solution to VRP with satellite facilities instance
References


References


