PROFIT MAXIMIZATION IN VEHICLE ROUTING PROBLEMS

Deniz Aksen
College of Administrative Sciences and Economics, Koç University

joint work with Necati Aras
Department of Industrial Engineering, Boğaziçi University
Intro: Capacitated Vehicle Routing

- **NP-hard** problem as a total distance / traveling cost minimization problem.

- Natural extension of the notorious traveling salesman problem (**TSP**) with...
  - Delivery vehicles (**homogeneous or heterogeneous fleet**) with limited capacity.
  - One or more central nodes (**single depot vs. multi-depots**) as origins of tours.
  - Each tour terminates at its own origin (**closed route**), or either at a different origin (**different depot**) or at a customer node (**open route**).
Natural extension of the notorious (TSP) with...
(cont.)

Customers with a known demand to be delivered
(linehaul customers) or
customers with a known supply to be picked up
(backhaul customers as in VRP with Backhauls.)

VRP with Simultaneous or Exclusive Pickups and Deliveries
(VRPPD)

A QoS (quality of service) guarantee in the form of two-sided time windows (TW), time deadlines (TD), maximum route duration or maximum route length.
Intro: Capacitated Vehicle Routing (cont.)

Natural extension of the notorious (TSP) with...

(cont.)

- The vehicle fleet size can be fixed (no costs associated with vehicles) or ...

- A vehicle acquisition cost (daily rental cost or discounted vehicle purchasing price) can be incurred.

**OBJECTIVES** subject to system constraints and QoS guarantees

- Minimization of total route length.
- Minimization of total operational costs.
CVRP Applications (cont.)

A **must** in e-commerce

- the e-fulfillment of online orders for perishable goods
- the realization of the last mile of delivery

A **quality of service** guarantee is time window restricted deliveries to customers’ dwellings.

![Diagram showing tours and delivery routes](image)

- Physical store serving also online customers
- Online (delivery) customer
CVRP Literature

A wide variety of real-world applications since its first introduction in 1959 (G.B. Dantzig and J.H. Ramser in *Mgmt Sci*) and since its most common solution method proposed in 1964 (G. Clarke and J.V. Wright in *Oper Res*)...

A few surveys and manuals of the VRP

1. **Computers & Operations Research**
The entire Volume 10, No. 2, 1983 special issue: 
*Routing and Scheduling of Vehicles and Crews – The State of the Art* by Lawrence Bodin, Bruce L. Golden, Arjang Assad and Michael Ball

2. **American Journal of Mathematical and Management Sciences**, the entire Volume 13, No. 3 and No. 4, 1993
CVRP Literature (cont.)

3. “Algorithms for the vehicle routing and scheduling problems with time window constraints”
   by Marius M. Solomon,

4. “Fleet Management and Logistics”
   by Teodor Gabriel Crainic and Gilbert Laporte (eds),
   Centre for Research on Transportation,

5. “A Computational Study of Vehicle Routing Applications”
   doctoral thesis by Jennifer L. Rich, RICE UNIVERSITY,
   Houston, May 1999, USA (Advisor: William J. Cook)
CVRP Literature (cont.)


10. “A guide to vehicle routing heuristics”


12. The Ultimate List of Vehicle Routing References at the Center for Traffic and Transport Research (CTT) of the Technical University of Denmark
   http://www.imm.dtu.dk/or/vrp_ref/vrp.html
13. **The VRP Web and the Networking and Emerging Optimization Group** at the **University of Málaga in Spain**
   [http://neo.lcc.uma.es/radi-aeb/WebVRP/](http://neo.lcc.uma.es/radi-aeb/WebVRP/)

14. **Branch Cut and Price Applications : Vehicle Routing Links**
   maintained by **Ted Ralphs** ([ted@branchandcut.org](mailto:ted@branchandcut.org)) at
   [http://branchandcut.org/VRP/](http://branchandcut.org/VRP/)

15. **Canada Research Chair in Distribution Management** with a collection of standard benchmark problems
    [http://www.hec.ca/chairedistributique/](http://www.hec.ca/chairedistributique/)
When $$$ enters the scene: The case of TSP


Traveling Salesman Problems with Profits (TSPs with Profits) are a generalization of the Traveling Salesman Problem (TSP) where it is not necessary to visit all vertices. With each vertex is associated a profit. The overall goal pursued is the simultaneous optimization of the collected profit and the travel costs. These two optimization criteria appear either in the objective function or as a constraint.

... Conclusions emphasize the interest of this class of problems, with respect to applications as well as theoretical results.

*Dominique Feillet, Pierre Dejax and Michel Gendreau*
TSP: Profit Maximization Case (cont.)

Alternative Objectives

- Minimize total vehicle distance whilst collecting a specified minimum amount of total profit from visited customers.

- Minimize total vehicle distance whilst serving a specified minimum number of customers.

- Maximize total profit collected from customers whilst a specified maximum total distance is traveled.

**JOINT OBJECTIVE FUNCTION**

MAXIMIZE Net Profit = Total Profit Collected — Total Cost of Traveling
OP and then TOP

- Unit demand
- No node-specific temporal constraints
- No vehicle capacity


MCP, and then MTMCP

MULTIPLE TOUR MAXIMUM COLLECTION PROBLEM (MTMCP)


Notation Scheme for VRP with Backhaul Opportunities

\( \alpha / \beta / \gamma \)

- Number of vehicles
- Backhaul Service Options [must or free]
- Precedence order between deliveries (linehaul) and pickups (backhaul) [prec or any]
Our Problem of Interest

- Single-depot capacitated vehicle routing problem with a flexible size fleet of homogeneous vehicles.
- Demand of all customers + Profit from all customers + coordinates of all customers and the depot known in advance with certainty.
- Time deadline + maximum route duration/length constraints are to be satisfied.

**JOINT OBJECTIVE FUNCTION**

\[ \text{MAXIMIZE Net Profit} = \text{Total Profit Collected} - \text{Total Cost of Deliveries} \]
VRPPTD: Profit Maximization Case (cont.)

**Mathematical Formulation (No TDs) – Alternative I**

max. \( \Pi = \sum_{i \in N \setminus \{0\}} p_i y_i - \text{UNITCOST} \times \sum_{i \in N} \sum_{j \in N, j \neq i} c_{ij} x_{ij} \)

s.t. \( \sum_{j \in N, j \neq i} x_{ij} = \sum_{j \in N, j \neq i} x_{ji} \) \( \forall i \in N \)

\( \sum_{j \in N, j \neq i} x_{ij} = y_i \) \( \forall i \in N \setminus \{0\} \)

\( \sum_{i \in N \setminus \{0\}} x_{0i} \geq \sum_{i \in N \setminus \{0\}} \frac{1}{Q} d_i y_i \)

\( \sum_{i \in S} \sum_{j \in S, j \neq i} x_{ij} \leq |S| - L_S \) \( \forall S \subseteq N \setminus \{0\} \) \( \wedge |S| \geq 2 \)

\( L_S = \left\lceil \frac{1}{Q} \sum_{i \in S} d_i \right\rceil \)

\( y_i, x_{ij} \in \{0, 1\} \)

Exponential number of Subtour Elimination Constraints
VRPPTD: Profit Maximization Case (cont.)

Mathematical Formulation (No TDs) – Alternative II

\[
\max \quad \Pi = \sum_{i \in N \setminus \{0\}} p_i y_i - \text{UNITCOST} \times \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ij}
\]

s.t.

\[
\sum_{j \in N \setminus \{i\}} x_{ij} = \sum_{j \in N \setminus \{i\}} x_{ji} \quad \forall i \in N
\]

\[
\sum_{j \in N \setminus \{i\}} x_{ij} = y_i \quad \forall i \in N \setminus \{0\}
\]

\[
\sum_{i \in N \setminus \{0\}} x_{0i} \geq \sum_{i \in N \setminus \{0\}} \frac{1}{Q} d_i y_i
\]

\[
u_i - u_j + Q \times x_{ij} \leq Q - d_j \quad \forall (i, j) \in N \setminus \{0\} \land i \neq j
\]

**OR:**

\[
u_i - u_j + Q \times x_{ij} + (Q - d_i - d_j) \times x_{ji} \leq Q - d_j \quad \forall (i, j) \in N \setminus \{0\} \land i \neq j
\]

\[
d_i \leq u_i \leq Q \quad \forall i \in N \setminus \{0\}
\]

\[
y_i, x_{ij} \in \{0, 1\}
\]

**Miller-Tucker-Zemlin** valid inequalities ([ACM](https://doi.org/10.1145/100236.100261), Vol. 7, 1960)

extended to the VRP by Kulkarni and Bhave ([EJOR](https://doi.org/10.1016/0013-7535(85)90018-3), Vol. 20, 1985),

**Lifted Version** by Kara, Laporte and Bektaş

([EJOR](https://doi.org/10.1016/j.ejor.2004.08.017), Vol. 158 (2004))
**VRPPTD: Maximize the NET PROFIT**

Solving VRPP and its time constrained variants

- Classical Heuristics preferred when solution time is more critical than solution quality.
  - (Parallel) Savings Algorithm by Clarke and Wright (1964)
  - Sweep Algorithm by Gillett and Miller (1974)
Solving VRPP and its time constrained variants

- Local Post Optimization (LPO) Procedures
  a.k.a. Local Improvement Heuristics (Parallel or First)

- 1-0 move (1-Opt) of *Golden, Magnanti and Nguyen* (1977)
- 1-1 exchange of *Waters* (1987)
- 3-opt of *Lin* (1965), of *Lin and Kernighan* (1973)
- 4-opt* of *Renaud, Laporte and Boctor* (1996)
- Or-opt of *Or* (1976)
VRPPTD: Maximize the NET PROFIT (cont.)

1-0 move LPO on the same route
VRPPTD: Maximize the NET PROFIT (cont.)

1-0 move LPO on 2 routes
VRPPTD: Maximize the NET PROFIT (cont.)

1-1 exchange LPO on the same route
VRPPTD: Maximize the NET PROFIT (cont.)

1-1 exchange LPO on 2 routes
VRPPTD: Maximize the NET PROFIT (cont.)

2-Opt LPO on the same route

\[ D \]

\[ i \]

\[ j \]

\[ i + 1 \]

\[ j + 1 \]

\[ D \]

\[ i \]

\[ j \]

\[ i + 1 \]

\[ j + 1 \]
2-Opt LPO on 2 routes
VRPPTD: Maximize the NET PROFIT (cont.)

Is customer \( i \) worth keeping (inserting) between nodes \( k \) and \( l \) ?

Triangular Inequality holds true even if distances are not perfectly Euclidean.

Predecessor of customer node \( i \): node \( k \)
Successor of customer node \( i \): node \( l \)

Marginal Profit of customer \( i \) = \( \pi_i - \text{Unit}_\text{Travel}_\text{Cost} \times (d_{ki} + d_{il} - d_{kl}) \)
Algorithm
iterative_Marginal_Profit_Analysis [iMPA]

**Initialization**
For the current set of routes $R$ compute the marginal profit value $M_{\pi_i}$ for each customer in the present set of customers $C$. Sort the $M_{\pi_i}$ values in *nondecreasing order* and obtain a sorted stack $M_{\pi_{[i]}}$.

**Step 2**
If $M_{\pi_{[0]}}$, the 1st *(thus: LOWEST)* marginal profit in the stack, which belongs to customer $i_{[0]}$ is positive then go to **Step 5**! O/w go to **Step 3**.

**Step 3**
Let $succ_i$ and $pred_i$ be the successor and predecessor nodes of customer $i_{[0]}$ on its current route $r_i$, resp. Delete $i_{[0]}$ from $r_i$. Discard it also from $C$. Update the $M_{\pi}$ values of $succ_i$ and $pred_i$ if they are customer nodes. Cancel route $r_i$ and update $R$ if $i_{[0]}$ was the only customer on it.

**Step 4**
Set $M_{\pi}$ value for $i_{[0]}$ to INFINITY such that it is put at the end of the stack. Restore the *nondecreasing order* of $M_{\pi}$ values in the stack and go to **Step 2**.

**Step 5**
Stop [iMPA].
Solving the VRPPTD with [iMPA]

**Initialization**
Solve the given problem instance as a total distance minimization VRPTD with the present (nondeleted) customers set $C$ in it. Apply to the total distance minimizing set of routes $R$ an LPO procedure of choice. In the resulting solution all customers in $C$ will be visited.

**Step 1**
For the current set of routes $R$ apply one round of the Marginal_Profit_Analysis [MPA].

**Step 2**
If [MPA] has not modified the current solution (i.e. [MPA] has not discarded any customers from $C$ and has not changed the current routes $R$),
then go to **Step 4**! O/w go to **Step 3**.

**Step 3**
Reset all routes and marginal profit values found in the previous steps. Delete the customer nodes that are found to be deleted in **Step 2** and go to **Initialization** to resolve the problem.

**Step 4**
Scan each final route and check if its net total profit value is positive. If not, then correct the route by using DROP heuristic until its net total profit becomes positive AND the disposal of the lowest marginal profit customer on it does not improve its net total profit anymore.
Flow Chart of solving VRPPTD with [iMPA]

START

Read problem data

Solve the problem using Parallel Savings Algorithm

Apply a series of 3 LPO heuristics (1-1, 1-0, 2-Opt)

Delete all routes, and discard customers with (−)ve Mπ, place again each remaining customer on a new dedicated route

Routes modified OR No MPA applied so far?

YES

Total profit increased (Positive Gain attained?)

YES

Apply one round of MPA

NO

STOP

Display solution

Inspect each route and apply [DROP] if its net total profit is negative.

NO
A Numerical Example solved: test0.txt

<table>
<thead>
<tr>
<th>CUST#</th>
<th>XCOORD.</th>
<th>YCOORD.</th>
<th>DEMAND</th>
<th>READY</th>
<th>DUE</th>
<th>SERVICE</th>
<th>UNIT</th>
<th>PROFIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100000</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>12</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>24</td>
<td>18</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>-8</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-16</td>
<td>12</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>-24</td>
<td>18</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>-8</td>
<td>-6</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>-16</td>
<td>-12</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>-24</td>
<td>-18</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>8</td>
<td>-6</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>16</td>
<td>-12</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>24</td>
<td>-18</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>40</td>
<td>30</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>-40</td>
<td>30</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>-40</td>
<td>-30</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>40</td>
<td>-30</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>30</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>-30</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>-30</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>100000</td>
<td>3</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

Uniform Vehicle Capacity: 6
A Numerical Example: test0.txt (cont.)

Coordinates of nodes

- Customers
- Depot
Before the MPA

- No customers lost : (0)
- No. required veh’s : 4
- Total Distance : 485.25
- Total Gross Profit : <480.00
- Total Net Profit : <-5.25>
After one sweep of the MPA

No customers lost : (2)
No. required veh’s : 4
Total Distance : |434.49|
Total Gross Profit : <450.00>
Total Net Profit : <15.51>
At the end of the MPA (2 sweeps)

No customers lost : (2)
No. required veh’s : 3
Total Distance : |421.89|
Total Gross Profit : <450.00>
Total Net Profit : <28.11>
Bremen, September 07th 2005

CPLEX 9.0 Optimal Solution
(148.9 sec on Pentium 4 3.40 GHz with 2 GB RAM)

No customers lost : (2)
No.required veh’s : 3
Total Distance : 367.53
Total Gross Profit : <405.00>
Total Net Profit : <37.47>
Solving the VRP Part with Simulated Annealing

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>312.81</td>
<td>3.687</td>
<td>YES</td>
<td>308.68</td>
<td>312.81</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>100.41</td>
<td>10.281</td>
<td>YES</td>
<td>87.68</td>
<td>91.81</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>108.53</td>
<td>0.400</td>
<td>YES</td>
<td>97.99</td>
<td>97.99</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>191.06</td>
<td>0.300</td>
<td>YES</td>
<td>181.92</td>
<td>181.92</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>213.14</td>
<td>0.410</td>
<td>YES</td>
<td>185.48</td>
<td>194.55</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>25.54</td>
<td>0.330</td>
<td>YES</td>
<td>0.03</td>
<td>25.54</td>
</tr>
<tr>
<td>7</td>
<td>15</td>
<td>26.97</td>
<td>73.890</td>
<td>NO</td>
<td>24.92</td>
<td>24.928</td>
</tr>
<tr>
<td>8</td>
<td>15</td>
<td>45.10</td>
<td>17.015</td>
<td>NO</td>
<td>21.86</td>
<td>21.86</td>
</tr>
<tr>
<td>9</td>
<td>15</td>
<td>17.23</td>
<td>6.406</td>
<td>YES</td>
<td>11.56</td>
<td>11.56</td>
</tr>
<tr>
<td>10</td>
<td>20</td>
<td>90.72</td>
<td>40.578</td>
<td>NO</td>
<td>90.72</td>
<td>90.72</td>
</tr>
<tr>
<td>11</td>
<td>20</td>
<td>247.23</td>
<td>~ 2hr</td>
<td>NO</td>
<td>230.45</td>
<td>230.45</td>
</tr>
<tr>
<td>12</td>
<td>20</td>
<td>186.61</td>
<td>~ 2hr</td>
<td>NO</td>
<td>108.65</td>
<td>156.71</td>
</tr>
</tbody>
</table>
Alternative Use of [iMPA] in a Greedy ADD-DROP Heuristic

**Initialization - ADD** *(starts with empty customer subset C)*

Preprocess customers by computing for each of them a *preferability coefficient* $PC_i$.

$$PC_i = \alpha \left( \frac{\pi_i}{D_i} \right) + \beta \left( \pi_i - UNITCOST \times \min_{j \in N \setminus \{0\}} \{d_{ij}\} \right) + \gamma \left( \frac{\max_{j \in N \setminus \{0\}} \{d_{0j}\}}{d_{0i}} \right) + \delta \left( \frac{\left| NEIGH(i, R) \right|}{n} \right)$$

Sort the $PC_i$ values in *nonincreasing order* and obtain a sorted stack $PC_{[i]}$ with the customer [0] being the most *preferable* one for addition to subset $C$. Set $i = 0$, BEST_PROFIT = 0.

**Step 1**

Add customer [i] of the stack $PC_{[i]}$ to $C$. On this current subset of customers, solve the VRPTD problem using Clarke-Wright Parallel Savings Algorithm with LPO to obtain the set of routes $R$.

**Step 2**

On the current set of routes $R$, apply the [iMPA] algorithm to find the best possible profit PROFIT realizable with $C$ and $R$. If PROFIT > BEST_PROFIT, then set BEST_PROFIT = PROFIT and record $C$ and $R$.

**Step 3**

Let $i = i + 1$. If $i = n$, then stop! Otherwise, go to **Step 2**.
Questions & Comments?

- Optimal Solution of test0.vrp
- Computational Results
- Algorithmic Flowchart
- No-Profit C-W Solution of test0.vrp