CIVIL-557

Decision Aid Methodologies In Transportation

Lab IX: Heuristics for the Hub Location Problem

Stefano Bortolomiol

Transport and Mobility Laboratory (TRANSP-OR)
École Polytechnique Fédérale de Lausanne EPFL

Overview

- Solution of the previous lab
- Introduction to the exercise: the hub location problem

Solution of the previous lab

Part 1 – Network Flow Model

Model

$$\min \quad \sum_{s\in S}\sum_{t\in S}\sum_{p\in P} c_{st}x_{stp} \qquad (1)$$

$$\text{s.t.} \quad \sum_{t\in S} x_{tsp} - \sum_{t\in S} x_{stp} = z_{sp} \qquad \forall s \in S, \forall p \in P \qquad (2)$$

$$q_{sp} = q_{sp-1} + I_{sp} + z_{sp} \qquad \forall s \in S, \forall p \in P \qquad (3)$$

$$q_{s4} = q_{s0} \qquad \forall s \in S \qquad (4)$$

$$q_{sp} \geq 3 \qquad \forall s \in S, \forall p \in P : p \leq 3 \qquad (5)$$

$$q_{sp} + I_{sp+1} \geq 0 \qquad \forall s \in S, \forall p \in P : p \leq 3 \qquad (6)$$

Part 1 – Network Flow Model

```
// --- OBJECTIVE FUNCTION ---

minimize
  sum(s in Stations, t in Stations, p in TimePeriods)
    Cost[s][t] * x[s][t][p];

// --- CONSTRAINTS ---

subject to
{
//BALANCING CONSTRAINTS

    // Compute the change in availability at each station due to repositioning
    forall (s in Stations, p in TimePeriods)
      sum(t in Stations) x[t][s][p] - sum(t in Stations) x[s][t][p]
      == z[s][p];

    // Station "bicycle conservation" constraints
    forall (s in Stations, p in TimePeriods: p >= 2)
      q[s][p] == q[s][p-1] + Imbalance[s][p] + z[s][p];
    forall (s in Stations)
      q[s][1] == Q_0[s] + Imbalance[s][1] + z[s][1];

    // Beginning of the day = end of the day
    forall (s in Stations)
      q[s][4] == Q_0[s];

    // Minimum number of bikes at each station at the beginning of each period
    forall (s in Stations, p in TimePeriods)
      q[s][p] >= 3;
    forall (s in Stations, p in TimePeriods: p <= 3)
      q[s][p] + Imbalance[s][p+1] >= 0;

}
```

TRANSP-OR

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Part 1 – Answers to questions

- Optimal solution
  - 1957
- Analysis of the flows
  - From lower to upper part of the city
  - Deterministic data – we are relocating only the minimum number of bicycles needed to satisfy the given demand
- How is the decision of relocating bicycles affected by our perfect knowledge of the future demand?
  - If we were to consider stochasticity, optimal solutions for the deterministic case would easily become bad solutions or even infeasible solutions.
  - A lot of research in transport optimization is focused on "robust" solutions of stochastic problems (i.e. solutions that are "good" under many different scenarios, such as demand fluctuations and supply disruptions).

Part 1 – Answers to questions

- Are the costs in the objective function realistic? How would you change them?
    - We assume each bicycle has the same transportation cost.
    - In practice, the cost function is not usually linear: the cost-per-unit decreases when the quantities increase.
    - More realistic formulation: fixed cost + per-unit cost

TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Part 2 – Routing Model

Model

$$\min \quad \sum_{i\in N}\sum_{j\in N}\sum_{p\in P} c_{ij} y_{ijp} + \sum_{s\in S}\sum_{t\in S}\sum_{p\in P} x_{stp}$$

$$\begin{aligned}
s.t. \quad & \sum_{t\in S} x_{tsp} - \sum_{t\in S} x_{stp} = z_{sp} && \forall s\in S, \forall p\in P && (8)\\
& q_{sp} = q_{sp-1} + I_{sp} + z_{sp} && \forall s\in S, \forall p\in P && (9)\\
& q_{s4} = q_{s0} && \forall s\in S && (10)\\
& q_{sp} \geq 3 && \forall s\in S, \forall p\in P : p\leq 3 && (11)\\
& q_{sp} + I_{sp+1} \geq 0 && \forall s\in S, \forall p\in P : p\leq 3 && (12)
\end{aligned}$$

$$\begin{aligned}
& \sum_{j\in N} y_{jip} = \sum_{j\in N} y_{ijp} && \forall i\in N, \forall p\in P && (13)\\
& \sum_{s\in S} y_{0sp} = 1 && \forall p\in P && (14)\\
& x_{stp} \leq C y_{stp} && \forall s\in S, \forall t\in S, \forall p\in P && (15)\\
& f_{ijp} \leq M y_{ijp} && \forall i\in N, \forall j\in N, \forall p\in P && (16)\\
& f_{ijp} \geq y_{ijp} && \forall i\in N, \forall j\in N, \forall p\in P && (17)\\
& f_{ijp} \leq \sum_{k\in N}\sum_{l\in N} y_{klp} && \forall i\in N, \forall j\in N, \forall p\in P && (18)
\end{aligned}$$

Part 2 – Routing Model

```
// --- OBJECTIVE FUNCTION ---

minimize
  sum(i in Nodes, j in Nodes, p in TimePeriods) Cost[i][j] * y[i][j][p] +
  sum(i in Stations, j in Stations, p in TimePeriods) x[i][j][p];


//ROUTING CONSTRAINTS

    // Flow conservation
    forall (i in Nodes, p in TimePeriods)
      sum (j in Nodes) y[j][i][p] == sum (j in Nodes) y[i][j][p];

    // The vehicle must leave from and return to the depot
    forall (p in TimePeriods)
      sum (s in Stations) y[0][s][p] == 1;

    // Capacity constraints
    forall (s in Stations, t in Stations, p in TimePeriods)
      x[s][t][p] <= Cap * y[s][t][p];

    // Subtour elimination
    forall (s in Stations, p in TimePeriods)
        sum (i in Nodes) f[s][i][p] - sum (i in Nodes) f[i][s][p] ==
        sum (i in Nodes) y[s][i][p];

    // Flow variables
    //  f[i][j][p] = 0 if the arc (i,j) is not used
    forall (i in Nodes, j in Nodes, p in TimePeriods)
      f[i][j][p] <=  nArcs * y[i][j][p];
    //  f[i][j][p] >= 1 if the arc (i,j) is used
    forall (i in Nodes, j in Nodes, p in TimePeriods)
      f[i][j][p] >= y[i][j][p];
    //  f[i][j][p] has as upper bound the number of arcs in the route for period p
    forall (i in Nodes, j in Nodes, p in TimePeriods)
      f[i][j][p] <= sum (k in Nodes, l in Nodes) y[k][l][p];
```

TRANSP-OR

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Part 2 – Comments

- We now model complete routes and not just flows.

- The structure of the costs changed.

- In the current formulation the objective function is increased by 1 for each bicycle carried over each arc.
  There are alternative way of formulating this:
  - Assigning an explicit parameter for the decision variables x_{ijp}
  - Consider only the total number of relocated bicycles in each period

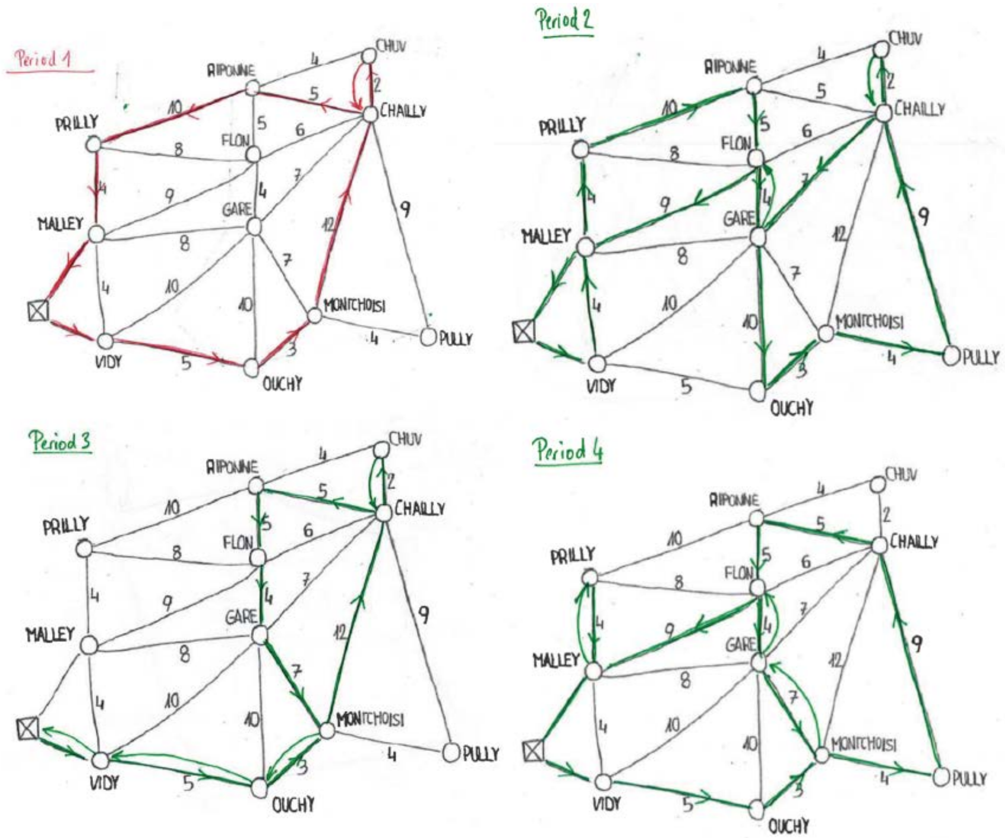
TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Part 2 – Answers to questions

- Optimal solution
  - 611 (228 without considering x_{ijp} variables in the objective function)

- Representation of the routes



TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Part 2 – Answers to questions

- What is the effect of vehicle capacity on the optimal solution?

Capacity	Optimal solution
35	606
40	606
50	605
60	605

Capacity	Optimal solution
25	622
20	642
15	697
10	1033

Part 2 – Answers to questions

- How can you further improve the model to make it more realistic?
  - Include travel times (relocation is not instantaneous) and consider congestion throughout the day
  - Allow for real-time reoptimization of the routes
  - Consider stochastic demands
    - Plan for uncertainty (aim at finding a robust solution)
    - Simulate the behavior of the system and estimate indicators (e.g. operation costs, level of service, number of unattended customers) for multiple days of operations
  - Consider multiple objectives and analyze trade-offs (Pareto optimality)
  - Give the option to use more than one vehicle (*Capacited Vehicle Routing Problem – CVRP*)
  - …and many more!

- In real life we always have "rich" problems which are instance-dependent.

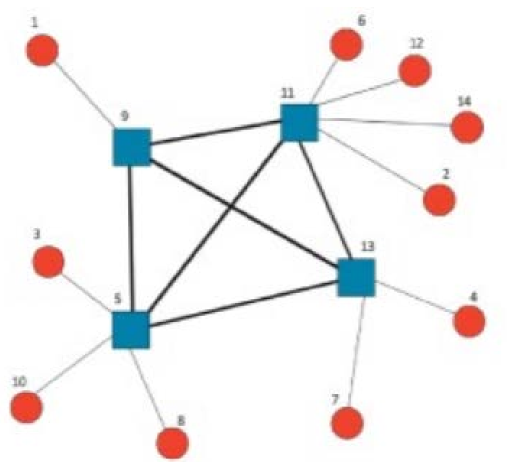
Exercise:
Heuristics for the Hub Location Problem

TRANSP-OR

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Hub location problem

- Commodities need to be shipped from an origin node to a destination node.
- In the network there are hubs where commodities coming from different origins are bundled together and shipped to other hubs which are close to their destinations.
- It is cheaper to route commodities through hubs (economies of scale = transport cost per unit decreases when the scale of operations increases)
- Objective function: $$\min \sum_{i \in N} \sum_{j \in N} h_{ij} \left(\sum_{k \in N} c_{ik} y_{ik} + \sum_{m \in N} c_{jm} y_{jm} + \alpha \sum_{k \in N} \sum_{m \in N} c_{km} y_{ik} y_{jm} \right)$$



TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Exercise tasks

- Write the objective function (`CalculateCost.m`)
  Hint: refer to the Basic p-Hub Location Model seen in the lecture

- Define different neighborhood structures (`NeighborhoodStructure.m`)
  Hint: think about the problem and be creative!

- Implement the simulated annealing metaheuristics (`SimulatedAnnealing.m`)

- Test your solution with different parameters and neighborhoods (`Main.m`)

- Report numerical and graphical results

Exercise questions

- **Neighborhoods**
  - Describe the neighborhood structures you have used and explain why you have chosen them.
  - How can the choice of the neighborhoods affect the quality of the algorithm?

- **Simulated annealing**
  - Explain the simulated annealing metaheuristic by analyzing the graph showing the values of solutions, accepted solutions and best solutions along the iterative process.
  - Compare the simulated annealing graph for two or more different neighborhood structures. In your case, does there seem to be any relevant difference in terms of quality between them? Whether yes or no, why do you think this happens?

Exercise questions

- **Heuristics**
  - In general, can you determine if a neighborhood structure (or a heuristic algorithm in general) is better than another? If yes, can you suggest a way to do it?
  - How "good" is the best solution found by your algorithm? What is the relation between the best solution found and the optimal solution?
  - Even if you do not know the optimal solution of the problem, is there a way to define a range within which it is contained (i.e. a lower bound and an upper bound)?

Assignment #9

- Send a zip file which contains your code and a document with the answers to the questions (add some figures as well).
- Submit to stefano.bortolomiol(at)epfl.ch by 8 pm next Monday.
- Group/individual work