An Empirical Study on Time-Dependent Delivery Route Planning

Qian An, Ugur Demiryurek and Cyrus Shahabi
Integrated Media Systems Center
University of Southern California

Introduction

- The majority of delivery companies design truck routes based on the assumption that travelling times on road segments are constant.
- In the real world, travelling time on each road segment is time-dependent, where travel time is determined by arrival time.

Time-dependent Route Planning with Time-window Constraint

- Starting from node 0, the delivery vehicle need to visit customer 1 to customer 4 and come back to node 0 as early as possible.
- The time window for each customer is [10,90]. If the vehicles arrives at a customer earlier than 10 or later than 90, there will be a high penalty cost.

When the traveling time between customer 3 and customer 4 increases by 40% during peak time, static planning leads to an arrival time of 92 at customer 4, which violates the time window.

Mixed-Integer Programming Formulation

- With an objective of minimizing total travelling time and penalty cost

\[
\begin{align*}
\min \quad & \sum_{ij} x_{ij} \cdot T_i + \sum_{ij} (y_i + \gamma_j) \cdot P \ \\
\text{subject to} \quad & T_r = 0 \\
& \sum_{j \in C_i} x_{ij} \leq X_i, \quad \forall i \in C, i \neq 0 \\
& \sum_{i,j} x_{ij} \leq L_i, \quad \forall i \in C \\
& \sum_{i,j} x_{ij} \cdot (T_j - T_i) \leq U_i, \quad \forall i \in C \\
& \sum_{i,j} x_{ij} \cdot (T_j - T_i) \leq L_i, \quad \forall i \in C \\
& \gamma_i = 0 \quad \forall i \in C
\end{align*}
\]

- Formulation inspired by Malandraki and Daskin (1992)
- Different objects could be used, for instance, maximizing the number of customers visited before 12pm, minimizing the number of delivery routes, etc.

Solution Strategy

- Time-dependent vehicle routing problem is NP-hard.
- Optimization software like CPLEX could solve small problem cases with less than 8 nodes and 20 time periods optimally.
- Heuristics have been proposed to solve large problem sets efficiently, including
  - Nearest-neighbor
  - Mathematical-programming-based heuristic
  - Genetic algorithm

Experiments

- Los Angeles road network with 304,162 edges, and 258 typical delivery customer zones
- Time-dependent edge travel-times are generated based on two years of historical data collected from 6300 traffic sensors. The sampling rate of the data is 1 reading/sensor/min.
- Sensor data is spatially and temporally aggregated by assigning interpolation points (for each 5 minutes) that depict the travel-times on the edges
- Static path and time-dependent shortest path between any two nodes was precomputed using our previous research method

Evaluation:
- For randomly selected sets of customers, find the optimal objective using CPLEX 12.3
- a) Time-dependent route planning and its corresponding object cost TT*
- b) Static planning with real-world travel time and its corresponding object cost TT
- c) report TT/TT*-1

Conclusion and Future Work

- Time-dependent planning reduces time window violations, increases the number of customers visited and reduces the number of delivery vehicles
- Time-dependent vehicle routing is NP-hard. Designing efficient heuristics is important to test on large empirical cases and provide good schedule for delivery companies.

Modeling

- Divide a day to multiple periods. The arrival time determines the travelling time between two nodes.

- Decision variables:
  - \( x_{ij} \) = 1 if link \((i,j)\) is utilized by a vehicle at time \( t \), \( x_{ij} = 0 \) otherwise
  - \( T_r \): time a vehicle leaves node \( i \)
  - \( y_i \): \( T_r \) if upper, lower time window at node \( i \) is violated; \( y_i \): \( T_r \) = 0 otherwise

- Parameters:
  - \( K \): number of available vehicles
  - \( P \): penalty cost of violating time window
  - \( B \): a large enough number
  - \( T_f \): traffic data updating frequency
  - \( S_j \): service time at node \( j \)
  - \( T_r \): travelling time on link \((i,j)\) at period \( t \)
  - \( U_i, L_i \): upper and lower time windows of customer \( i \)
  - \( C \): set of depot and customers (deposit is node 0)
  - \( M \): set of time periods