IEOR 151 – Lecture 22
Vehicle Routing Problem

1 Problem Formulation

In the vehicle routing problem, there are a set of depots, vehicles, and delivery locations, and the problem is to optimally design routes for the vehicles from the depots to delivery locations. To formally define the version of the problem that we will consider in this class, suppose that there is an undirected graph $G = (V, E)$ with vertices $V = \{v_0, v_1, \ldots, v_n\}$ and edges $e_{ij} \in E$ between vertices v_i and v_j. *One important facet of this model is that each vertex is connected by an edge.* Without loss of generality, we will assume that the depot is located at vertex v_0, and there are delivery locations (e.g., buildings, cities, etc.) at the remaining vertices. Furthermore, the edges e_{ij} are weighted by d_{ij}, which represents the distance between vertices v_i and v_j. The maximum number of vehicles is n, which would be the situation in which exactly one vehicle is assigned to each delivery location. Each vehicle has a maximum capacity of goods G, and each vehicle has a maximum distance D that it can travel. Finally, each delivery location v_i for $i \geq 1$ has a demand value w_i.

The vehicle routing problem is to find least-cost vehicle routes so that

1. Each delivery location is visited exactly once by exactly one vehicle;
2. All vehicles start and end at the depot;
3. Side constraints on maximum vehicle capacity and travel distance are satisfied;

2 Integer Linear Programming Solution

There are a number of different solutions to this problem, depending upon the size of the problem and also the side constraints (other constraints such as on times windows for delivery are possible by modifying the problem formulation). One solution is to use an integer linear program (ILP), but the weakness of this approach is that there are many constraints and requires special numerical approaches with set partitioning and column generation. As a side note, this approach is quite similar to the cycle-weight formulation for kidney exchanges.

Let $C(G, D)$ be the set of all feasible routes, in which a single route $r \in C(G, D)$ is such that a vehicle does not visit the same delivery location more than once, the demand weight of all delivery locations on the route is less than G, the total (round-trip) distance of the route is less than D, and the route starts and ends at the depot vertex v_0. For any $r \in C(G, D)$, define $L(r)$ to be the total (round-trip) distance of the route. Lastly, define x_r to be a binary decision variable the denotes whether a route is or is not in the optimal solution. Then, we can formulate the problem as the following ILP

$$\begin{aligned}
\min \sum_{r \in C(G,D)} & L(r) x_r \\
\text{s.t.} \sum_{r \in C(G,D)} & 1(v_i \in r) \cdot x_r = 1, \forall i \geq 1 \\
& x_r \in \{0,1\}, \forall r \in C(G,D),
\end{aligned}$$

where $1(v_i \in r)$ is an indicator function that is 1 whenever $v_i \in r$ and is 0 otherwise.

3 Clarke and Wright's Savings Algorithm

Solving this problem exactly can be difficult, and so a number of heuristics have been developed. One of the conceptually simplest heuristics is Clarke and Wright's Savings Algorithm. The algorithm proceeds as following:

1. Make n routes: $v_0 \to v_i \to v_0$, for each $i \geq 1$;
2. Compute the savings for merging delivery locations i and j, which is given by $s_{ij} = d_{i0} + d_{0j} - dij$, for all $i, j \geq 1$ and $i \neq j$;
3. Sort the savings in descending order;
4. Starting at the top of the (remaining) list of savings, merge the two routes associated with the largest (remaining) savings, provided that:
   (a) The two delivery locations are not already on the same route;
   (b) Neither delivery location is interior to its route, meaning that both notes are still directly connected to the depot on their respective routes;
   (c) The demand G and distance constraints D are not violated by the merged route.
5. Repeat step (3) until no additional savings can be achieved.

4 More Information and References

The material in the first two sections of these notes follows that of the journal paper G. Laporte, "The Vehicle Routing Problem: An overview of exact and approximate algorithms," *European Journal of Operational Research*, vol. 59, pp. 234–358, 1992. The material in the last section of these notes follows the course textbook "Service Systems" by Mark Daskin.