`POPMUSIC` for the World Location Routing Problem

Adriana C.F. Alvim[1], Éric D. Taillard[2]

[1] Department of Applied Informatics,
Federal University of the State of Rio de Janeiro (UNIRIO),
Avenida Pasteur 458, Rio de Janeiro, RJ, 22.290-240, Brazil.
adriana@uniriotec.br

[2] Department of Industrial Systems, HEIG-Vd,
University of Applied Sciences of Western Switzerland,
Route de Cheseaux 1, Case postale, CH-1401 Yverdon, Switzerland.
eric.taillard@heig-vd.ch

Abstract

The goal of this work is to show that the `POPMUSIC` methodology can be applied to large location routing problems (*LRP*). The size of instances currently considered in the litterature is limited to few dozens of customers and few depots, even for approaches based on metaheuristics. The size of practical problem can be much larger, typically several thousands of customers and hundreds of depots. This implies that the methods developped up to now cannot be used directly for practical problems. However, these methods can be embedded in `POPMUSIC` allowing much larger instances to be treated. Indeed, in `POPMUSIC` methodology, the solution improvement with a basic metaheuristic such as tabu search is not the part that contributes most to the algorithmic complexity. This work deals with an instance 10,000 larger than those commonly treated in the litterature. Such an instance can be treated by a standard laptop in less than one CPU day.

1 The World Location Routing Problem

The Location Routing Problem is a well known optimization problem [2]. Several versions of this problem exist, and we have considered the following one in order to generate instances based on the World *TSP* (http://www.tsp.gatech.edu/world). Let N be a set of n customers. Each customer has a non-negative demand q_i. The distance c_{ij} between customers i and j is known. Set N is also the set of n possible depot locations with unlimited capacity each and where is based an unlimited vehicle fleet, each vehicle having a given capacity Q. The cost of opening a depot is O, the cost of a route is its length and each vehicle has a fixed cost. The objective is to find which depots should be opened and which routes should be constructed to minimize the total cost (costs of depots, routes and vehicles) such that: (i) each demand q_i must be served by a single vehicle which performs a single route; (ii) each route starts and ends at the same depot; (iii) the total demand of a route does not exceed Q. The data of the World *TSP* is given solely by the longitude and latitude of each city. To create a World *LRP*, we have considered the (x, y, z) cartesian coordinates (in meters) obtained by projection on the ellipsoid of the GRS80 Geodetic Reference System (http://en.wikipedia.org/wiki/GRS_80). The demand of customer i, located at position (x_i, y_i, z_i) is given by a pseudo-random value: $q_i = 1 + (107\lfloor x_i \rfloor + 97\lfloor y_i \rfloor + 163\lfloor z_i \rfloor) \; modulo \; 30$. Various World *LRP* instances can be created by modifying this pseudo-random generator as well as the depot opening cost O and vehicle capacity Q.

2 Heuristic for World *LRP*

Our approach used to solve the World *LRP* is presented in Algorithm 1. `POPMUSIC` works by locally optimizing sub-problems build on a given solution. First of all, an initial *LRP* solution must be generated. This first phase is the most complex part of the method since it is not so easy to build a decent solution with an algorithmic complexity lower than $O(n^2)$. Indeed, optimizing a *LRP* solution (Line 8) can be performed in quasi-linear time, as it is for labeling the objects of a map [1]. The approach for generating an initial solution follows roughly the lines of [3] for tackling large centroid clustering problems. The

---

Data: Set N of customers; sample size s; vehicle capacity Q,depot opening cost O.
Result: Solution for the World *LRP*.
1 Instance reduction and customer sampling;
2 Creation of big clusters;
3 Decomposition of big clusters into clusters;
4 Assignement of a depot for each cluster and solving a *TSP* for each cluster;
5 Splitting large tours;
6 Merging depots of several tours;
7 Build MDVRP solution by satisfying vehicle capacity constraints;
8 Improve MDVRP solution with `POPMUSIC`;

Algorithm 1: Phases of World *LRP*.

---

process starts by decomposing the problem into sub-problems of not so large size (around 10^4 cities). The result of this phase is a set of big clusters (Line 2). Each big cluster is then decomposed into clusters of customers that can (almost) be delivered by a single vehicle tour (Line 3). At the end of this DEC phase, a local optimization phase (LOPT) starts, following the lines of `POPMUSIC`. The algorithmic complexity of the DEC + LOPT phase, as initially proposed in [3], is quasi linear in nm where m is the number of clusters (in our case the depots) and n is the number of entities (in our case, cities or customers). For practical problems, the number of customers that can be assigned to a vehicle tour can be considered as a constant. In practice, the number of customers in a tour is limited to few dozens. This means that the number of vehicles needed for delivering n customers is in $O(n)$. So, the DEC + LOPT phase is in $O(n^2)$, which is impractical for $n \gg 10^4$. Therefore, a first sampling phase is added to speed-up the process (Line 1). A sampling procedure is used to create a reduced set of customers N_s of the original problem N. The size of N_s is limited to the order of 10^4. The sampling procedure just selects customers randomly, uniformly. This reduced set is decomposed into a small number of big clusters by solving heuristically a p-median problem (with $p = \sqrt{|N_s|}$). In order to create clusters containing roughly the same number of customers, a Lagrangean relaxation of a capacitated p-median problem (*CPMD*) is solved. The idea is to give the same capacity to each cluster and allowing capacity violation but with a penalty (Lagrangean multiplier). The *CPMD* is solved with a procedure based on a Candidate List Search (`CLS`) [3]. Once the position of the p centers have been determined, all n customers of the original problem are assigned to the closest center. The decomposition of big clusters into clusters of customers that can be delivered by a single vehicle is also done by solving a *CPMD* with a Lagrangean relaxation. This is done with the same procedure than for the creation of big clusters. Since the capacities are considered as soft constraints, the target capacity of clusters is set to the vehicle capacity diminished by 5%. By doing so, there are few clusters that have a capacity larger than the actual vehicle capacity. Next, in Line 4, the customers of a cluster are arranged in a tour by solving a *TSP* instance. This is performed with a local search based on the 2-opt neighborhood. The customer at the centroid of each cluster is selected to be a depot. There are few tours that are very long, due to the dispersion of the cities in very sparse region of the world. In case the length l_i of a tour is larger than the cost O of opening a depot, the tour is tried to be split (Line 5). Next, in Line 6, a procedure is used to merge depots. After the merging procedure, there are tours violating the vehicle capacity. These tours are split while keeping them attached to the depot. As a result, we have a feasible *VRP* solution for each depot (Line 7).

Finally, in Line 8, the `POPMUSIC` frame is applied to optimize the solution. The `POPMUSIC` frame works as follows : The solution is decomposed into parts, in our case, a part is a vehicle tour. Then, a *seed* part is selected, together with its r closest parts for creating a sub-problem R. Sub-problem R, which is in our case a *MDVRP* is tentatively optimized. If R has been improved, all its parts are considered for functioning as *seed* part for creating further sub-problems. Otherwise, the *seed* part is discarded for creating new sub-problems. Initially, all parts are considered for creating sub-problems and the process stops when all parts have been discarded. The sub-problem optimization is done with a basic tabu search procedure.

	Procedure	CPU Time Seconds	# Depots fixed	# Depots not fixed	Route cost	# Vehicles
1	Instance reductions	7.36	255			
2	Creation of big clusters	44.69	255	141		
3,4	Creation of clusters and *TSP*	8,602.17	255	95,289		
5	Tour splitting	26.56	710	95,698	9,066,166,720.63	
6	Depots merging	704.43	710	23,971	9,789,992,078.84	
7	Build MDVRP instance	0.39	710	23,971	10,377,889,016.03	109,058
8	POPMUSIC MDVRP	74,222.63	710	23,971	9,255,604,273.39	107,177
	TOTAL	83,608.23				

Table 1: Preliminary results for an instance with 1,904,711 customers.

3 Computational Results

Computational experiments were performed on a Processor Intel Core 2 Duo 2.53GHz with 3.2GB of memory. Yet, only the concept of the above technique has been tested. So, we have only preliminary results for a single instance with $|N| = 1,904,711$ customers that was created by setting $O = 100,000$ and $Q = 300$. Preliminary results are shown in Table 1. First and second columns indicate the corresponding procedure of Algorithm 1; the next column indicates the CPU time taken by the procedure; the next two columns show the number of depots (fixed depots are those with only one customer); next column shows the sum of the length of all routes and the last column the total number of vehicles used in the current solution. The initial instance reduction found 255 cities without city at a distance less than O. These cities are considered as fixed depots and removed from the remaining of the process. The creation of 141 big clusters took 44.69 seconds with a sample size of $s = 20,000$ customers. It took almost 2.4 hours to decompose all big clusters into 95,289 clusters by solving *CPMD* instances. Splitting long tours took only 26.56 seconds and leads to the creation of 455 new fixed depots and 409 other depots delivering more than one customer. The total length of the tours is about 21% above the best solution known to the World *TSP*. Note however that the World *TSP* length is neither a lower bound nor an upper bound to the World *LRP*. The next procedure merged and transformed the 95,289 depots in 23,971 depots in around 12 minutes. This procedure increased the tours length by about 8%. Splitting the tours that are overloaded to create a feasible *LRP* solution with 109,058 vehicles tours increased the length by another 6%. POPMUSIC methodology improved the initial *MDVRP* solution by more than 1 million of kilometers (11%) and 1,881 vehicles in almost 21 hours.

Our ongoing work is to design a better optimization procedure for optimizing sub-problems, especially by relocating depots instead of solving *MDVRP* instances. Other instances will be considered for presenting extensive computational experiments.

Acknowledgment

References

[1] Adriana C.F. Alvim and Éric D. Taillard. Popmusic for the point feature label placement problem. *European Journal of Operational Research*, 192(2):396–413, January 2009.

[2] Leon Cooper. Location-allocation problems. *Operations Research*, 11:331–343, 1963.

[3] Éric D. Taillard. Heuristic methods for large centroid clustering problems. *Journal of Heuristics*, 9:51–73, 2003.