THE TABU SEARCH HEURISTIC METHOD FOR A MULTI VEHICLE DISTRIBUTION AND ROUTING PROBLEM

Olgierd DZIAMSKI

Abstract. A good quality distribution plan plays a very important role in the transportation process. The vehicle planning and scheduling problem for many cities and vehicles is known to be NP-hard. Real-life vehicle routing problems for furniture distribution impose additional requirements on goods and pallet placements on trucks. The aim of this paper is to present a combination of modern heuristics to generate a feasible distribution plan for real furniture distribution.

1. Introduction

This paper shows the implementation of heuristic algorithms for solving real-word distribution planning and loading activities for the furniture industry. There are almost unlimited numbers of different types of distribution models used for delivering furniture to the final customers. The simplest one is the Travel Salesmen Problem (TSP). Conceptually, it is very simple: the travelling salesman must visit every city in his territory, every single one and then return home covering the shortest distance. A more complex model is the Multi Vehicle Routing Problem (MVRP). This problem requires a set of vehicles to visit a set of customers from one depot. There are several important problems that must be solved in real-life. The side constraints, such as the maximum truck capacity, different numbers of goods and form of packing required by every visit have to be taken into consideration. This paper introduces a tabu search (TS) heuristic with additional loading heuristics as a good technique for solving complex multi-vehicle routing problems for furniture distribution in a reasonable time.

2. Literature review

The vehicle routing problem (VRP) has generated a considerable amount of research over the last three decades, as can be seen in the bibliography by Laporte and Osman (1995). The best existing optimal algorithms for the VRP appear to be those of Hadjiconstantinou et al. (1995) [7], and can rarely solve problem cases involving less then 135 customers. The best heuristics for the VRP appear to be the Tabu search-based algorithms of Taillard (1993), Osman (1993) and Gendreau et al. (1994). The first class of heuristics was developed between 1960 and 1990 and can be called classical heuristics. The metaheuristics belong to the second class, whose growth has occurred in the last fifteen years. The most widely known classical heuristic for the VRP are Clarke and Wright's (1964) savings algorithm, the swap mechanism (Gillett and Miller (1974)), and cluster-first route-second methods (Fisher and Jaikumar (1981). The first heuristic algorithm is based on a constructive idea, in which the vehicle route is created in one phase. More advanced heuristics contain the post-optimization procedure, like r-opt (Lin (1956)) or 2-opt (Protein and Rousseau (1995)).

Most classical heuristics work in a descent mode until a local optimum is reached. Most standard construction and improvement procedures in use today belong to the metaheuristics class. These methods, based on neighborhood search, perform a limited exploration of the search space and generally produce good quality solutions within modest computing times. Over the last fifteen years, the tabu search metaheuristic have been applied to the VRP by several authors. Some of the first TS algorithms did not yield impressive results, but subsequent implementations were much more successful. One of the first successful implementation of the TS algorithm, call Taburoute, was proposed by Gendreau et al. (1994) [4]. He defines the neighborhood structure as the solutions that can be reached from the current one by removing a vertex from its current route, and inserting it into another route. In addition, he makes it possible to examine a solution that may be unfeasible with respect to the capacity or maximum route length constraints. In addition, Taburoute does not use a tabu list, but random tabu tags. Whenever a vertex is moved from one route to another, its reinsertion into the first route is forbidden until some certain iteration. Later, the concept of Taburoute was improved by Taillard [5] by constructing a more sophisticated neighborhood structure, like the λ-interchange generation mechanism. The Xu and Kelly's [6] tabu search implementation contains swaps of vertices between tow routes for global repositioning of some vertices into other routes.

The VRP can also be successfully solved by genetic algorithms, which operate on a population of encoded solutions called chromosomes. It is based on applying a two parents chromosomes crossover operation, which generates two offspring. In the next step the worst elements of the populations are replaced with the offspring. Nowadays we can see some research works on learning mechanisms, hybrid algorithms and complex techniques for searching very large neighborhoods.

3. Building model

Increasing competition in the furniture market is forcing companies to deliver their product to customers in a smaller volume than that of the maximal capacity of all available trucks. In every case if companies use their own or hired fleet, the cost of delivery is proportional to the distances covered by all the vehicles engaged in distribution activities. For this reason, a good quality distribution plan should serve many clients with each truck. Furthermore, the sequence of visits made by every vehicle are restricted by many side constraints. The most common ones are related to vehicle capacity, route duration and maximum ride time. In addition to these, furniture products are very often distributed on pallet units, which have to be properly loaded on every truck. All in all, the right distribution plan for all trucks will be feasible when all simple and complex constraints will be fulfilled.

3.1. Distribution model

In this paper, the concept of calculating a furniture distribution plan for many different packing types to all clients for available trucks will be explained. The distribution model of the multi-vehicle routing problem is composed of one depot, many clients and the number of available trucks. The model contains many real-life constraints typical for furniture distribution. All the routes have to start and end in one depot. Each truck has its maximum carrying capacity calculated in kg, length, width, height and price for one kilometer. In addition, all the trucks can only make one trip in a calculated distribution plan. Each activity in the depots and in the cities is modeled as a visit. Clients can choose one of the forms of packing furniture directly on the floor or on pallets. The vehicle starts from its own visit to the depot and, after making many visits on route, returns back to the depot.

For this model, the goal is to minimize the global distribution cost for all the trucks, so that each visit is performed exactly once. The potential distribution plan is illustrated in figure 1.

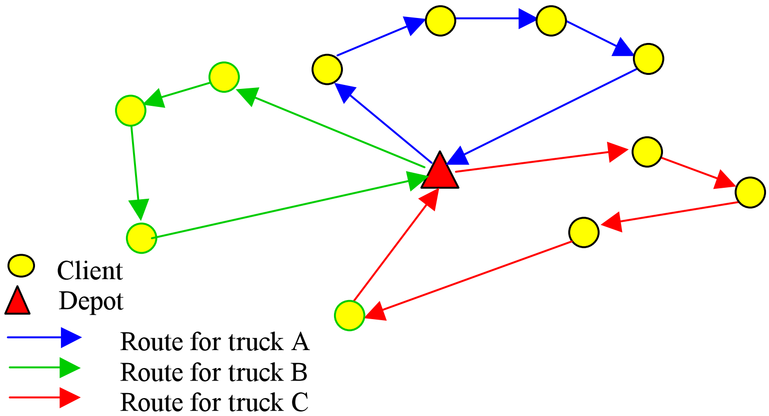
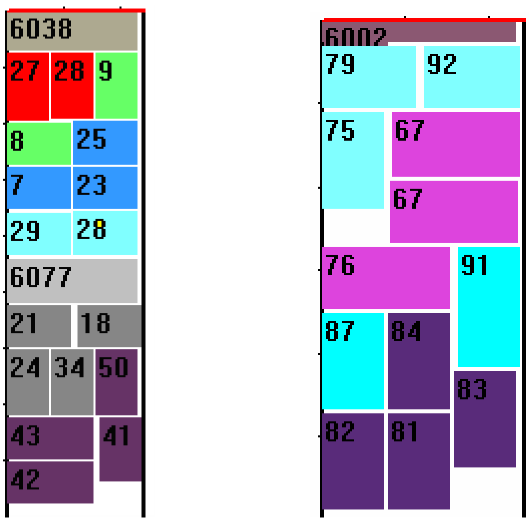


Figure 1. Multi-vehicle Routing Problem

In this representation, the search space 'S' for distributing furniture consists of all possible permutations where the capacity constraints of every truck are not exceeded. If we are concerned with a symmetric of every route, the maximal search space is (n-1)!/2. In fact, while building every route the feasibility of loading furniture on the floor and pallets has to be calculated. For the first problem, call vehicle routing problem (VRP) (Backer et al. 1997; Solomon 1987), there is proof that it is classified as NP-hard by Tsang et al. 1999. The second are knapsack problems and it is also classified by many researchers like Donald L. Kreher, Douglas R. [2] as belonging to NP-hard problems. In this paper, the combination tabu search heuristic for solving vehicle routing problems with loading heuristics has been explained as a powerful solution for real-life distribution goods.

3.2. Loading model

Furniture can be loaded directly on the floor of a truck or in pallets. In both cases goods have to be placed in a proper location that allows them to be unloaded from the rear part of the vehicle. In figure 2, we can see two types of truck loaded with pallets and furniture. When the truck makes the first visit, all the pallets for this visit will be directly accessed and unloaded. In figure 2, for truck 1 all the palettes or furniture for a certain visit are drawn in the same color. Then, pallets 41, 42, 43 and 50 will be unloaded on the first visit. After this operation, the pallets for the second visit can be easily accessed. The process can be repeated till the last visit of the truck. For the third and the eighth visits (orders: 6077 and 6038) the tuck is loaded directly on the floor with furniture. All trucks are loaded in the inverse sequence to the places they visited. This phenomenon requires recalculating the loading plan for trucks in every case, when there are any changes in the sequence or number of visits. The loading model keeps loading constraints to support the loading and unloading schema described above. I use a single pass heuristic, which calculated the best loading schema for pallets or furniture for the deepest unfilled space. If the space cannot be filled with any other pallets or furniture, it is merged with another to create a bigger one. In addition, the loading heuristic uses predefined loading patterns in order to calculate a high quality loading plan for whole truck. The process is repeated till all the pallets and furniture are loaded. If any pallet or furniture is left for any visit on the truck route, the loading plan is unfeasible.



Truck 1: 245cm * 600cm **Truck 2: 245cm * 900cm**

Figure 2. truck loaded with furniture

4. Exact and Heuristics Search Techniques

The simplest approach to solve a NP-hard optimization problem is to list all the feasible solutions, evaluate their objective function values and choose the best. This approach is called the exact method. It is unusable in practice because of the vast number of possible solutions to many vehicle problems.

In the early days of combinatorial optimization, the distribution problems could be solved by using integer variables and a method generally known as “Branch & Bound” (B&B). This efficient enumeration scheme which avoids complete enumeration of solutions can significantly reduce a search space. Despite enormous research done to improve performance of this method, the computational complexity of these algorithms grows exponentially with the size of the problem. As a result of this, Branch and Bound codes usually do not scale well to large instances of vehicle routing or to knapsack problems Naddef, D., and Rinaldi, G. [8]. Because of the complexity of these problems, the research effort has focused on heuristics. These techniques sacrifice the proof of optimality for solutions and instead focus on finding good, near optimal solutions at a reasonable computational cost. Unfortunately, it may not be possible to state how close to optimality a particular heuristic solution is.

5. Searching for a Solution

The tow-phase approach has been used to solve complex multi-vehicle furniture distribution problems. The first phase consists of generating a solution that calculates the first feasible solution for distribution and loading goods. This preliminary solution is generated with nearest additional heuristic. The method begins from adding the depot to the partial route for a given truck. In the next step the closest city to the end of the current partial route is added. Whenever new city is added, the separate sub model calculates a feasible loading plan for all the pallets and furniture for all the visits on the truck route. If it is not possible to find such a city without violating constraints, the current partial route is closed. The whole process is repeated for another empty truck. The heuristic finishes with success if all the cities are added to the routes. In phase two, the solution is improved by using a local search and iterative improvement techniques.

5.1. Local Search

Local Search, also referred to as Neighborhood Search, is the basis of many heuristic methods for combinatorial optimization problems. It is a simple iterative method based on the trial and the error activity for finding good approximate solutions. The combinatorial optimization problem can be defined by a pair (S, g). S is the set of all feasible solutions where all routes satisfy the distribution and loading constraints and g is the objective function that maps each element s in S to a total distribution cost. In our furniture distribution model the goal is to find the solution s in S that minimizes the objective function g. The problem is stated as:

$$\min g(s), s \in S \tag{1}$$

A neighborhood N for the problem instance (S, g) can be defined as a mapping from S to its powerset:

$$N : S \to 2^S \tag{2}$$

N(s) is called the *neighborhood* of s and contains all the solutions that can be reached from s by a single *move*. The *move* is an operator, which transforms one solution to another with small modifications. A solution x is called a local minimum of g with respect to the *neighborhood* N if:

$$g(x) \le g(y), \forall y \in N(x) \tag{3}$$

Local search is the procedure of minimizing the cost function g in a number of successive steps in *each* of which the current solution x is being replaced by a solution y such that:

$$g(y) < g(x), y \in N(x) \tag{4}$$

A local search algorithm begins with a solution generated in phase one and ends up in a local minimum where no further improvement is possible. This method is called hill climbing, or greedy algorithm, which replaces the current solution with the solution that decreases the most goal function g, which is the total distribution cost. The local minimum is the main problem with local search. To eliminate this problem, I use the tabu search heuristic as a main control mechanism for guiding the whole search process for route building and truck loading. This approach allows local search to be guided beyond the local minimum.

5.2. Building Neighborhood

For furniture distribution model, I used four types of move operators. The neighborhood generated by these operators is classified in two groups: those that modify only one route are called intra-route neighborhoods; those that make changes between routes are called inter-route neighborhoods. The intra-route neighborhoods are generated by a 2-Optimal operator or 4-Optimal operator. In the 2-Optimal, two arcs in a single route are cut and reconnected to improve the total cost of the route. For figures 3 and 4 the cost is proportional to the length of the route. Figure 3 illustrates this process. The move eliminates the crossing by destroying two arcs and creating two new arcs, if all capacity and

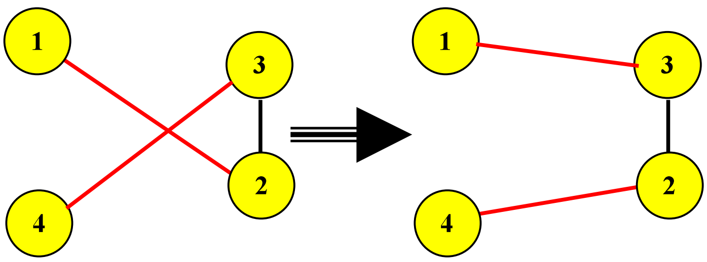


Figure 3. 2-Optimal Neighborhood

loading constraints are still satisfied. The resulting route is shorter, and thus less costly. The 4-Optimal operator works in a similar way to 2-Optimal. Four arcs in a single route are cut and four new arcs are reconnected to improve the total cost of the route.

For inter-route neighborhoods, the exchange and relocate operators are implemented. The first one, two visits of two different routes, swap places if all capacity and loading constraints are still satisfied. In the second one, a visit is inserted in another route if all capacity constraints are still satisfied. The move destroys three arcs and creates three new arcs. This process is shown in figure 4.

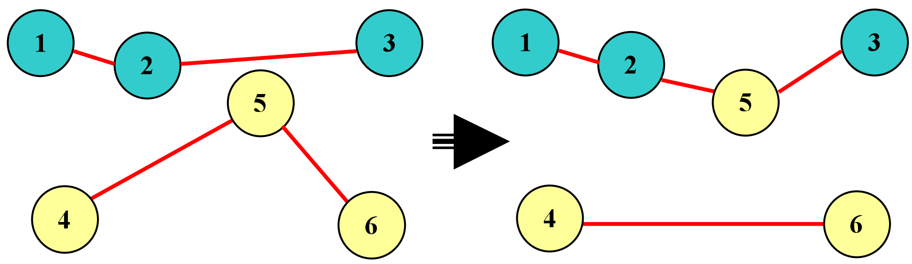


Figure 4. Relocate Neighborhood

6. Tabu Search

Tabu Search (TS) has been developed by Glover [1]. TS is a heuristic that combines a local search process with a number of anti-cycling rules which prevent the search from getting trapped in a local minimum. The Tabu Search works similarly to an ordinary local search, proceeding iteratively from one solution to another until it reaches a certain number of iterations. For the current solution, every iteration neighborhood $N(x) \subset S$ is built by all move operators. The best move is selected to transform the current solution to a new one. For this solution another neighborhood $N^*(y) \subset S$ is built. A key aspect of tabu search is the use of a special memory structure which serves to determine $N^*(y)$ and hence to organize the way in which the space is explored. For furniture distribution, I use recency-base and long-term memory structure to guide search process.

6.1. Recency-Base memory

Tabu search performs the best improvement local search, selecting the best move in the neighborhood but only amongst those not characterized as *tabu*. All moves are stored on recency-based memory. They are kept for certain number of iterations to forbid reverse moves. In this tabu search variant for the distribution model, any type of move operator is classified as tabu only if all the added edges by selected move operator are on the tabu list. If not all of the added edges are on the tabu list, then the candidate move is not classified as tabu. All the deleted edges of the selected move are placed on the list. If the list is full, the oldest elements of the list are replaced by the new deleted edges information.

6.2. Frequency-Base memory

Diversification strategy in the distribution model is designed to drive the search into new regions. The frequency-base memory was implemented in order to go to a more fruitful region that is different than the one being currently explored. To force the search into a different extent the choice rules is modified by adding penalty function:

$$MoveValue' = MoveValue + f * Penalty \tag{5}$$

Penalty value is a function of frequency measure which consists of ratio, whose numerator represent the number of times edge *(i,j)* or *(j,i)* has been added while improving moves. The denominator represents the maximum numerator value. Factor *f* corresponds to a desire for more diversification. In distribution model, nodes that are added to the solution more frequently are penalized more heavily to encourage the choice of moves that incorporate other nodes.

References

[1] Fred Glover, Manuel Laguna. Tabu Search. *Kluwer Academic Publisher 1997*

[2] Donald L. Kreher, Douglas R. Stinson Combinatorial Algorithms Generation, Enumeration, and Search, *1999 by CRC Press LLC*

[3] Gendreau, M., Hertz, A., Laporte, G., 1992. New insertion and post optimization procedures for the traveling salesman problem. Operations Research 40, 1086–1094.

[4] Gendreau, M., Hertz, A., Laporte, G., 1994. A tabu search heuristic for the vehicle routing problem. Management Science 40, 1276–1290

[5] M. Gendreau, G. Laporte, C. Musaraganyi and E. D. Taillard, A Tabu Search Heuristic for the Heterogeneous Fleet Vehicle Routing Problem, Computers & Operations Research 26 (1999) 1153–1173.

[6] Xu, J., Kelly, J.P., 1996. A network flow-based tabu search heuristic for the vehicle routing problem. Transportation Science 30, 379-393.

[7] Hadjiconstantinou, E., Christofides, N., Mingozzi, A., 1995. A new exact algorithm for the vehicle routing problem based on q-path and k-shortest paths relaxations. Annals of Operations Research 61, 21–43.

[8] Naddef, D., and Rinaldi, G. (2002). Branch-and-cut algorithms for the capacitated VRP. In: Toth, P., and Vigo, D. (eds), The Vehicle Routing Problem, pages 53–84. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia.