1 Introduction

In many practical contexts, loading constraints have a strong impact on the way in which delivery routes can be organized. This important dimension of routing operations has been realized since a few years and several authors have recently addressed variants of classical routing problems in which loading constraints are accounted for. In this presentation, we consider a routing problem in which a single vehicle must move a given number of objects from a set of pickup locations to a set of delivery locations subject to complex loading constraints. Each object is loaded at a specified pickup point and must be delivered to a specified delivery location. Objects are assumed to have a standard width and can thus be placed in a given number of “stacks” in the vehicle. Once loaded, objects cannot be moved to other stacks or resequenced in their stack, and only the (current) last object of each stack is available for delivery. Another constraint is that the total length of the objects making up each stack at any time must not exceed the length of the vehicle.

This problem is an extension of the one in which items are packed in a LIFO fashion on a single vehicle that performs pickup and delivery operations. This problem has been addressed recently by Carrabs et al. [1], who proposed a Variable Neighborhood Search approach for it, and in [2], where an exact method based on branch-and-bound was presented. The use of multiple stacks to model two-dimensional loading constraints on...
vehicles was introduced by Petersen [3] who considered a problem in which a vehicle must first pickup a group of unit-width objects and load them in several stacks, travel a long distance, and then deliver these objects subject to loading constraints. This problem differs significantly from ours in the fact that all pickup operations are grouped together at the beginning of the route, while we allow mixing pickup and delivery operations.

2 An “Adaptive Large Neighborhood Search” Heuristic

The solution method that we propose is based on the “Adaptive Large Neighborhood Search” heuristic that Pisinger and Ropke [4] applied very successfully to a large range of routing problems, including pickup and delivery ones. Pisinger and Ropke’s approach consists in systematically deleting a large fraction of the elements of the current solution using adapted destruction operators and then “reconstructing” a new solution from the remainder using adapted reconstruction operators. The selection of the specific destruction and construction operators to be applied in any given iteration is made randomly and depends on probabilities that are adjusted dynamically through a learning mechanism to reflect the good or bad performance of the various operators.

Our heuristic is made up of three phases. An initial solution is first constructed by sequentially inserting requests in the route and the stacks. Then, the main “Adaptive Large Neighborhood Search” phase is performed. It follows the procedure presented in [4], but uses four destruction operators that respectively remove randomly a given number of requests, a group of geographically close requests, a group of requests picked up sequentially, and the requests assigned to a given stack; two insertion operators, based respectively on minimum cost and minimum regret criteria, are applied to reconstruct complete solutions. It should be noted that the new solution derived at a given iteration replaces the current one only if it is an improving one or if it passes a probabilistic test as in simulated annealing. A third phase exploits the fact that a solution to the problem is defined both by the assignment of pickup and delivery points to stacks and by the exact route followed by the vehicle. For a given feasible set of assignments of pickup and delivery points to stacks, one may derive several feasible routes of different length. Finding the optimal route for a given assignment can be performed by a dynamic programming procedure of complexity $O(n^k)$, where $n$ is the number of requests to be transported and $k$ is the number of stacks in the vehicle.

3 Computational Results

The proposed procedure was tested on four sets of benchmark instances. The two first sets are for the single-stack variant of the problem and were proposed in [1] and [2]. The first set is made up of 42 instances having between 12 and 375 requests, while the second group of 63 instances contains from 9 to 21 requests. The third benchmark is the one proposed by Petersen [3]. It is made up 60 instances of the (DTSPMS) having a 3-stack vehicle and 12, 33 or 66 requests. DTSPMS instances can easily be transformed into instances of our problem by suitably defining distances between the depots and pickup and delivery locations. Finally, the fourth benchmark is a new set of instances based of the 28 first instances of [1]; three new instances having between 2 to 5 stacks
are defined for each original one and allow testing different features of the solution procedure.

References

2. F. Carrabs, R. Cerulli, and J.-F. Cordeau, An Additive Branch-and-Bound Algorithm for the Pickup and Delivery Traveling Salesman Problem with LIFO or FIFO Loading, INFOR, 45, 223-238 (2007)