



**HAL**  
open science

## An adaptive large neighborhood search for a full truckload routing problem in public works

Axel Grimault, Nathalie Bostel, Willy Lambert, Fabien Lehuédé

► **To cite this version:**

Axel Grimault, Nathalie Bostel, Willy Lambert, Fabien Lehuédé. An adaptive large neighborhood search for a full truckload routing problem in public works. *Odysseus 2015*, May 2015, Ajaccio, France. hal-01168035

**HAL Id: hal-01168035**

**<https://hal.science/hal-01168035v1>**

Submitted on 25 Jun 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# An adaptive large neighborhood search for a full truckload routing problem in public works

Axel Grimault<sup>1</sup>, Nathalie Bostel<sup>2</sup>, Willy Lambert<sup>3</sup>, Fabien Lehuédé<sup>1</sup>

<sup>1</sup> LUNAM Université, Ecole des Mines de Nantes, IRCCyN, Nantes, France  
{axel.grimault, fabien.lehuede}@mines-nantes.fr

<sup>2</sup> LUNAM Université, IUT de Saint-Nazaire, IRCCyN, Saint-Nazaire, France  
nathalie.bostel@univ-nantes.fr

<sup>3</sup> Luc Durand Travaux Publics, Pruillé, France  
w.lambert@durandtp.fr

**Keywords:** *Truck routing and scheduling, full truckload, resource synchronization, ALNS.*

## 1 Introduction

This paper presents a truck routing and scheduling problem faced by a public works company. It consists of optimizing the collection and delivery of materials between sites, using a heterogeneous fleet of vehicles. These flows of materials arise in levelling works and construction of roads networks. As the quantity of demands usually exceeds the capacity of a truck, several trucks are needed to fulfill them. As a result, demands are split into full truckloads. A set of trucks routes are needed to serve a set of demands sharing a set of resources, available at pickup or delivery sites, which can be loaders or asphalt finishers in our application cases. Thus, these routes need to be synchronized at each resource. We propose an Adaptive Large Neighborhood Search (ALNS) to solve this problem. This approach is evaluated on real instances from a public work company in France.

## 2 Problem statement

We consider an operational transportation problem for daily transport demands. Materials are shipped directly from source sites to destination sites. According to the capacity of the heterogeneous fleet of trucks and because mixing products in a truck is not allowed, all transports in our application are considered as full truckloads. An elementary transport of one material with one truck is called a request. Several full truckload requests have to be served by a set of heterogeneous vehicles routes to fully satisfy a demand. Trucks with various capacities are used, therefore the number of requests needed to satisfy a demand is not known before solving.

We distinguish two kinds of demands that are subject to two kinds of resource synchronization constraints. *Scheduled demands* are synchronized with route construction and have to be delivered starting from a given time without interruption. On the contrary, *unscheduled demands* are planned along a time period with more flexibility according to wide time windows. The service of demands sharing the same resource cannot overlap. Each resource can carry out only one operation at a time

(unitary capacity). The need to schedule the service of several vehicles on a given resource is called resource synchronization [1]. The optimization problem consists of designing a set of minimum cost routes to fulfill the demands, satisfying capacities and resource synchronization constraints.

A two-phase method algorithm is proposed in [3] to address this problem. The first phase splits demands into full truckload transportation requests while the second phase constructs a set of routes of minimum cost using a routing and scheduling heuristic. This paper focus on solving the vehicle routing problem.

### 3 Problem formulation

The quantity of products of a demand  $n \in N$  is denoted by  $q_n$ . Each request  $r \in R$  has a pickup point  $p_r$  and a delivery point  $d_r$ . The set of pickup points is denoted by  $P$  and the set of delivery points  $D$ . The service (pickup or delivery) at a point  $i \in P \cup D$  must start within the time window  $[e_i, l_i]$ . The duration of the service at a point  $i \in P \cup D$  is denoted by  $s_i$ . A truck  $k \in K$  has a capacity  $Q^k$ . The cost of using arc  $(i, j)$  by truck  $k$  is denoted by  $c_{ij}^k$ . The considered horizon time is a day. The time at which vertex  $i$  is served is denoted by  $h_i$ . A truck is allowed to arrive at vertex  $i$  before the start of the vertex time windows, in this case, it has to wait before starting the service.

The problem consists of assigning each request  $r$  to a compatible truck  $k$ , designing the routes of the trucks in order to satisfy synchronization on resources and the time windows for each request. The objective is to minimize the cost function which includes a fixed cost per vehicle, a cost of the duration of routes and a distance-traveled cost. This problem is referred as a Full Truckload Pickup and Delivery Problem with Resource Synchronization (FT-PDP-RS). A similar problem, dealing with the delivery of ready mix concrete, has been discussed in [8] and solved with a hybrid approach relying on Variable Neighborhood Search. A representation of the studied problem and its solution is presented in Figure 1.

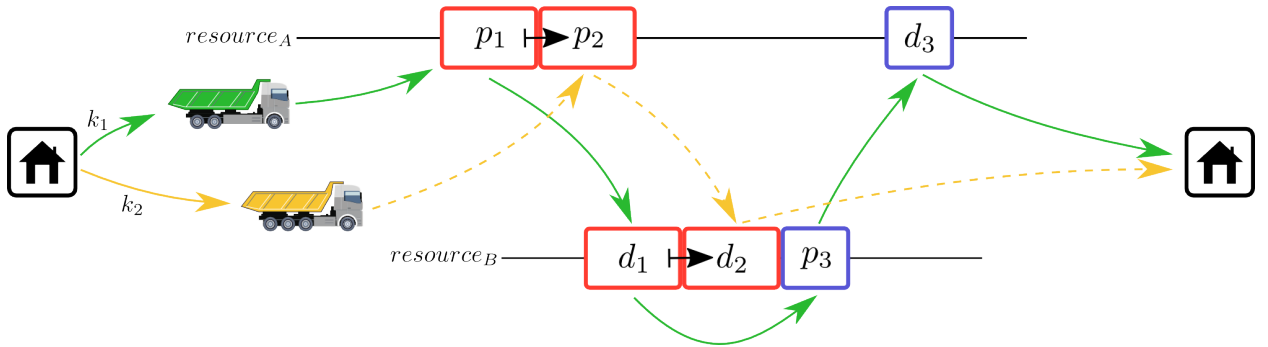


Figure 1: This figure illustrates the problem and the solution for two *scheduled* requests 1 and 2 and one *unscheduled* request 3. Two trucks  $k_1$  in solid line and  $k_2$  in dashed line visit requests starting and finishing at their depots. Synchronized pickups of requests 1 and 2 are sharing resource A with delivery of request 3. Synchronized deliveries of request 1 and 2 are sharing resource B with pickup of request 3. Note that truck  $k_1$  has to wait for the completion of the delivery of request 2 before starting the pickup of request 3.

### 4 Optimization approach

A metaheuristic based on Adaptive Large Neighborhood Search (ALNS) is proposed to solve the FT-PDP-RS. This metaheuristic uses competitive destroy and repair operators to improve the incumbent

solution. ALNS has been introduced in [6].

ALNS uses iteratively destroy operators that removes requests from the routes and repair operators that re-insert these requests. An adaptive layer selects the best operators among a number of destroy and repair operators to improve a solution according to their past performance. An acceptance criteria from simulated annealing is used to accept a better solution and escape from local minimum. The algorithm stops when a given number of iterations has been reached. The following paragraphs present the characteristics of these operators and the feasibility evaluation of a solution.

#### 4.1 Destroy and Repair operators

Our algorithm implements destroy operators that remove requests from routes and resources. The global framework of these operators is to remove a number  $q$  of requests. Removed requests are stored in a *request bank*. Classical destroy operators from the literature have been implemented. *Random Destroy Operator* simply removes  $q$  requests randomly. *Worst Destroy Operator* removes  $q$  requests that cause the biggest detour, see [6]. *Distance-related Destroy Operator* is inspired from the heuristic proposed by [9] and adapted in [6]. For each requests pair  $i$  and  $j$ , this operator evaluates the relatedness  $r_{ij}^{dist} = \frac{1}{2}(d_{p_i, d_j} + d_{d_i, p_j})$ . It removes a set of requests that are closed to each other. *Time-related Destroy Operator* removes requests served around the same time using relatedness  $r_{ij}^{time} = |h_{p_i} - h_{p_j}| + |h_{d_i} - h_{d_j}|$ . Moreover, problem specific operators have been implemented. *Route Destroy Operator* selects randomly a route and removes all of its requests. *Resource Destroy Operator* selects randomly a resource and removes all of its requests. The expected advantage of the resource removal is to reorder in a better way the requests sharing the same resource.

Repair operators are specific to our problem. Because the problem deals with resource synchronization, requests have to be inserted both in routes and on resources. We develop a general insertion heuristic which embed the selection of a request to insert, an operator to insert in routes, an operator to insert on resources. The insertion on a route is always performed before the insertion on a resource. *Cheapest Insertion* inserts a request that generates the smallest increases in distance. *Earliest Truck Appending* inserts a request at the end of the route that implies the earliest service time among all possible routes. On resources, we propose two operations to schedule requests. *Earliest feasible* schedules a request at its earliest feasible time on a resource while *Latest Feasible* schedules a request at its latest feasible position time. The main criterion for insertion is to minimize the increase of distance.

#### 4.2 Feasibility of the solution

The consideration of the synchronization between resources introduces new challenges. A large number of routes are linked due to synchronization of requests in routes and resources. Inserting a request on a solution may impact the feasibility of the time windows of its successors in both routes and resources. To control the feasibility, the forward time slack  $F_i$  at vertex  $i$ , which represents the amount of time the beginning of service can be postponed without violating the feasibility of the solution, from [7] have been extended for the Pickup and Delivery of Transfers (PDPT) in [5] and extended to the Two-Echelon Multiple-Trip Vehicle Routing Problem with Satellite Synchronization in [2]. Based on the previous work, we develop an algorithm to check in constant time the feasibility of the insertion of a request in a solution with respect to all temporal constraints.

## 5 Conclusion

The proposed algorithm is tested on real instances from the literature and on real case instances from a public work company. Our algorithm improves solutions from [4]. The associated numerical experiments will be presented on the conference. The algorithm is currently being integrated in a more global decision support system designed for a public work company.

## Acknowledgment

This work is part of project ORLoGES (Optimisation du Réseau Logistique en Génie Civil, avec les aspects Economiques et Sociétaux) funded by "Caisse des Dépôts" and supported by Nov@log.

## References

- [1] Michael Drexler. Synchronization in vehicle routing - A survey of VRPs with multiple synchronization constraints. *Transportation Science*, 46(3):297–316, March 2012.
- [2] Philippe Grangier, Michel Gendreau, Fabien Lehuédé, and Louis-Martin Rousseau. An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization. *Les cahiers du CIRRELT*, (CIRRELT-2014-33):17p, 2014.
- [3] Axel Grimault, Fabien Lehuédé, and Nathalie Bostel. A two-phase heuristic for full truckload routing and scheduling with split delivery and resource synchronization in public works. In *Logistics and Operations Management (GOL), 2014 International Conference on*, pages 57–61, June 2014.
- [4] Patrick Hirsch. Minimizing empty truck loads in round timber transport with tabu search strategies. *International Journal of Information Systems and Supply Chain Management*, 4(2):15–41, 2011.
- [5] Renaud Masson, Fabien Lehuédé, and Olivier Péton. Efficient feasibility testing for request insertion in the pickup and delivery problem with transfers. *Operations Research Letters*, 41(3):211 – 215, 2013.
- [6] Stefan Ropke and David Pisinger. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4):455–472, 2006.
- [7] M. W. P. Savelsbergh. Local search in routing problems with time windows. *Operations Research*, 4:285–305, 1985.
- [8] Verena Schmid, Karl F Doerner, Richard F Hartl, and Juan-José Salazar-González. Hybridization of very large neighborhood search for ready-mixed concrete delivery problems. *Computers & Operations Research*, 37(3):559–574, March 2010.
- [9] Paul Shaw. Using constraint programming and local search methods to solve vehicle routing problems. In Michael Maher and Jean-Francois Puget, editors, *Principles and Practice of Constraint Programming - CP98*, volume 1520 of *Lecture Notes in Computer Science*, pages 417–431. Springer Berlin Heidelberg, 1998.