

FREDERIC SEMET
UNIV. LILLE, CNRS, CENTRALE LILLE

frederic.semet@centralelille.fr



7TH SCIENTIFIC SEMINAR

Branch-and-Price approaches for a Fleet Composition Problem

Abstract

In this presentation, we address a Fleet Composition Problem (FCP) faced by Last-Mile Delivery Service Companies (LMDSC) that are middlemen between e-commerce companies and carriers.

LMDSCs organize transportation services for e-commerce companies and take advantage of higher volumes to mutualize more efficiently the transportation part. LMDSCs do not manage their fleet but have contracts with local carriers. Carriers propose different types of vehicles (bicycles, motorcycles, cars, and vans) with different transportation costs per km. One day in advance, LMDSCs have to decide how many vehicles of each type will be needed to cover the transportation demand.

One of the main characteristics of the problem addressed is that the demand is not known a priori, due to the inherent uncertainty of this type of activity. However, the demand can be approximated. Additional constraints such as vehicle capacities, the maximum working time have to be taken into account. The FCP, we tackle, consists in determining the minimum cost vehicle fleet to cover the demand while satisfying such side-constraints. The total cost is computed as the sum of handling costs and traveling costs.

First, we developed a compact integer linear formulation for the FCP described above. We then derived an extended formulation from it by applying a Dantzig-Wolfe decomposition. In this model, each request has to be assigned to a vehicle while minimizing the total cost. The pricing problem per type of vehicle approximates the performed route. We developed a branch-and-price algorithm as well as a diving heuristic to solve this model. The branch-and-price method includes several features such as:

- 1) preprocessing techniques to set some variables in the subproblem;
- 2) heuristics to solve the subproblem;
- 3) Specific rules to generate several columns for a given type of vehicle at each iteration;
- 4) an adapted branching scheme.

To assess the efficiency of the proposed approaches, we solved real-life instances provided by our industrial partner. The diving heuristic performs quite well since the gaps range between 1.87% and 2.25% while the CPU times are between 342 and 603 seconds.