



HAL
open science

Exact algorithms for the picking problem

Lucie Pansart, Nicolas Catusse, Hadrien Cambazard

► **To cite this version:**

Lucie Pansart, Nicolas Catusse, Hadrien Cambazard. Exact algorithms for the picking problem. ROADEF 2017, Feb 2017, Metz, France. hal-01799036

HAL Id: hal-01799036

<https://hal.science/hal-01799036>

Submitted on 24 May 2018

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.

Exact algorithms for the picking problem

Lucie Pansart, Nicolas Catusse, Hadrien Cambazard

Univ.Grenoble Alpes, CNRS, G-SCOP, 38000 Grenoble, France

{lucie.pansart,nicolas.catusse,hadrien.cambazard}@g-scop.grenoble-inp.fr

Keywords : *picking, integer programming, Steiner TSP, compact formulation, cuts*

1 Introduction

Warehouse Management Systems (WMS) are dedicated to help companies to reduce their logistical costs. They currently use heuristic to solve the picking problem, which is aimed at finding the shortest tour to collect every items of a picking list in a warehouse. Given (a) a warehouse with v vertical aisles containing the slots where the items are stored and blocks separated by h horizontal cross-aisles, (b) a depot and (c) a picking list giving the location (block and aisle) of n items to collect, the objective is to minimize the tour of the order picker, beginning and ending at the depot and collecting every ordered items. Figure 1 gives an example of an instance. It is a particular case of the Traveling Salesman Problem (TSP).

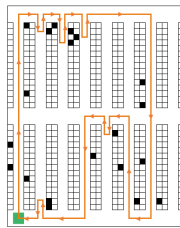


FIG. 1: An example of a warehouse with 2 blocks (= 3 cross-aisles) and 8 aisles. The depot is the bigger square. Black slots represent items to collect. The edges show the shortest tour for this instance.

Current exact algorithms to solve this problem use dynamic programming [2, 4]. We focus here on Mixed Integer Linear Programming (MILP). It is well-known that two important factors for a MILP formulation to be effective are the space of the solutions and the strength of the linear relaxation. This work proposes a compact formulation and improvements to strengthen the linear relaxation. Moreover, our algorithm has the advantage to be flexible, meaning side constraints specific to a company can be easily added. The algorithm proposed can be integrated to WMS, which usually contain a linear solver.

2 Modeling

We model the problem with a Steiner graph $G(V = R \cup T, A)$. The set R of required vertices represents the items and the depot: they *must* be visited in the tour. The set T of Steiner vertices represents intersections between vertical and horizontal aisles: they *can* be visited in the tour. An arc corresponds to an existing path between two vertices in the warehouse. Each arc (ij) has a cost d_{ij} , equal to the distance between the vertices i and j . We define the variables, for each arc (ij) : $x_{ij} = 1$ if the tour uses the arc (ij) , 0 otherwise. The objective is to minimize the cost of the tour $\sum_{(ij) \in A} d_{ij}x_{ij}$. We use a formulation using the flow principle: the order picker leaves the depot with n units of a commodity and delivers one unit each time

he picks an item. This formulation uses a polynomial number of constraints but suffers from its compactness: the lower bound given by its linear relaxation is lower than the exponential formulation lower bound [3].

Our contribution is made on several sides: we create a preprocessing that reduces the size of the Steiner graph and adds cuts to the MILP formulation to increase the linear relaxation. The preprocessing keeps the minimal number of vertices and arcs used to solve the problem. Cuts are inspired by the exponential formulation. They are defined as (S, \bar{S}) , partitioning the warehouse into two subsets such as each subset contains at least one required vertex. It implies that there must be at least one arc going from S to \bar{S} (and one from \bar{S} to S). We therefore add the constraints $x(S : \bar{S}) \geq 1$. To keep the compactness of our formulation, we add a polynomial number of cuts, relevantly chosen based on the warehouse structure.

3 Results

To solve our formulation, we used CPLEX 12.6. We apply it on 120 instances, split into 12 classes, of a benchmark proposed by Theys *et al.* [5]. We compared our algorithm with three other exact approaches: dynamic programming [2], TSP solver Concorde [1] and a compact TSP formulation (i.e., with a complete graph and not a Steiner graph) also solved with CPLEX.

The first result is that adding preprocessing and cuts improves the Steiner formulation: the gap between the linear relaxation and the optimal solution moves from 41.7% to 1.1%. In average, our algorithm is slower than Concorde and the dynamic programming. However, on many instances the computing time is reasonable: smaller than 1 second for 5 classes and smaller than 30 seconds for 4 other classes. Finally our formulation solves more instances than the TSP formulation in the time limit: in 30 minutes, our algorithm solved 110 instances while the TSP formulation solved only 77 instances.

Note also that the adaptation of our preprocessing for Concorde leads to the resolution of each instance of the entire benchmark in less than 5 minutes, which is a big improvement on this benchmark [5].

4 Conclusion and perspectives

This work proposes a new approach to solve the picking problem: a compact and flexible MILP formulation based on a Steiner graph and improvements to reinforce this formulation, easy to implement in a WMS embedding a MILP solver. On an academic point of view, our contribution was to solve the entire benchmark very efficiently without the need of heuristics.

Tracks for further work are mainly to get stronger reasoning on the structure of the data to get more efficient algorithm. It would be moreover desirable to obtain an industrial benchmark to test our algorithm on real instances. Finally, it may be interesting to link this problem with other warehouse issues such as storage policy or batching.

References

- [1] D. Applegate, R.E. Bixby, V. Chvátal, and W. Cook. Concorde. Available at: www.tsp.gatech.edu.
- [2] Hadrien Cambazard and Nicolas Catusse. Fixed-parameter algorithms for rectilinear steiner tree and rectilinear traveling salesman problem in the plane. *arXiv preprint arXiv:1512.06649*, 2015.
- [3] Adam N Letchford, Saeideh D Nasiri, and Dirk Oliver Theis. Compact formulations of the steiner traveling salesman problem and related problems. *European Journal of Operational Research*, 228(1):83–92, 2013.
- [4] H Donald Ratliff and Arnon S Rosenthal. Order-picking in a rectangular warehouse: a solvable case of the traveling salesman problem. *Operations Research*, 31(3):507–521, 1983.
- [5] Christophe Theys, Olli Bräysy, Wout Dullaert, and Birger Raa. Using a tsp heuristic for routing order pickers in warehouses. *European Journal of Operational Research*, 200(3):755–763, 2010.