The Active-Passive Vehicle-Routing Problem, Part I: Solution by Column Generation

Christian Tilk\textsuperscript{1}, Nicola Bianchessi\textsuperscript{1,2}, Michael Drexl\textsuperscript{1}, Stefan Irnich\textsuperscript{1}, and Frank Meisel\textsuperscript{3}

\textsuperscript{1}Chair of Logistics Management, Gutenberg School of Management and Economics, Johannes Gutenberg University Mainz (JGU Mainz) – Germany
\textsuperscript{2}Department of Information Engineering, University of Brescia, Italy – Italy
\textsuperscript{3}Christian-Albrechts-Universität zu Kiel (CAU) – Germany

Abstract

This paper presents a branch-and-price algorithm for the exact solution of the active-passive vehicle-routing problem (APVRP). The APVRP covers a range of logistics applications where pickup-and-delivery requests necessitate a joint operation of active vehicles (e.g., trucks) and passive vehicles (e.g., loading devices such as containers or swap bodies). The problem supports a flexible coupling and decoupling of active and passive vehicles at customer locations in order to achieve a high utilization of both resources. Accordingly, the operations of the vehicles have to be synchronized carefully in the planning. The contribution of the paper is twofold: Firstly, we present an exact branch-and-price algorithm for this class of routing problems with synchronization constraints. To our knowledge, this algorithm is the first such approach that considers explicitly the temporal interdependencies between active and passive vehicles. The algorithm is based on a non-trivial network representation that models the logical relationships between the different transport tasks necessary to fulfill a request as well as the synchronization of the movements of active and passive vehicles. Secondly, we contribute to the development of branch-and-price methods in general, in that we solve, for the first time, an ng-path relaxation of a pricing problem with linear vertex costs by means of a bidirectional labeling algorithm. Computational experiments show that the proposed algorithm delivers improved bounds and solutions for a number of APVRP benchmark instances. It is able to solve instances with up to 76 tasks, 4 active, and 8 passive vehicles to optimality within two hours of CPU time.