

# A multi-directional local search metaheuristic for a bi-objective dial-a-ride problem

Yves Molenbruch<sup>1,2</sup>, Kris Braekers<sup>1,2</sup>, An Caris<sup>1,2</sup>

<sup>1</sup> Hasselt University, Research Group Logistics, Agoralaan Gebouw D, 3590 Diepenbeek, Belgium

<sup>2</sup> Research Foundation Flanders (FWO), Egmontstraat 5, 1000 Brussels, Belgium  
{yves.molenbruch, kris.braekers, an.caris}@uhasselt.be

**Keywords:** *dial-a-ride, demand-responsive transportation, vehicle routing, bi-objective optimization, multi-directional local search, variable neighborhood search*

## 1 Introduction

A **dial-a-ride system** [4] is an application of demand-dependent, collective people transportation. Each user requests a trip between an origin and a destination of choice, to which a number of service level requirements are linked. The service provider attempts to develop efficient routing schedules, respecting these requirements and the technical constraints of a pickup and delivery problem [12]. The balancing of human and economic perspectives involved in solving such a dial-a-ride problem (DARP) explains why these systems are particularly useful for organizing quality-oriented, but efficient transportation for users having special needs, such as door-to-door transportation for elderly and disabled [17]. Since demand for dial-a-ride systems is generally increasing, service providers need efficient planning algorithms to safeguard quality and cost efficiency.

The **objective** of this paper is to solve a bi-objective problem variant in which the conflicting interests of users and service providers are both incorporated as an objective. Contrary to single-objective methods, which usually minimize operational costs while ensuring a minimum quality level imposed by the service level requirements, the fundamental nature of the DARP is emphasized by explicitly minimizing user inconvenience. To this end, a multi-directional local search (MDLS) metaheuristic [16] is developed, in which a variable neighborhood search (VNS) framework is embedded to guide the local search part. Initially, this metaheuristic is applied on a problem having standard characteristics [3], but its applicability can be extended by incorporating additional real-life features, as will be illustrated.

The remainder of this paper is organized as follows. In section 2, related publications in recent literature are briefly reviewed and compared with the contributions of the MDLS strategy applied in this paper. In section 3, the characteristics of the problem under consideration are formalized. The solution method is presented in section 4, after which computational tests are discussed in section 5. Opportunities for future research are identified in section 6.

## 2 Literature review

Multi-objective problems are often approached using a weighted-sum objective function. However, the at first sight readily interpretable objective value hides trade-offs between different goals and requires a priori weight choices. Regarding local-search based metaheuristics for the multi-objective DARP, three alternative approaches can be distinguished in **recent literature**. Lehu  d   et al. [6] integrate large neighborhood search (LNS) in a multi-criteria utility model based on the Choquet integral, considering the relative importance of different criteria and the interactions between them. Still, the importance of these criteria has to be defined in advance. Besides, users are assumed to have a limited number of common destinations, which is exploited by the LNS operators. Parragh et al. [13] consider standard problem characteristics [3] and aim at approximating the Pareto set of non-comparable solutions, among which the decision maker can choose afterwards. VNS is applied repeatedly, minimizing a weighted sum of total distance traveled and mean user ride time for multiple weight combinations. Additional non-dominated solutions on the linking path between known solutions are discovered using path relinking. Paquette et al. [10] also keep a set of solutions, but increase the number of objectives and add heterogeneous users and driver-related constraints. Tabu search is combined with a reference point method, summing distances between an intermediate solution and an ideal point over all objectives. The objective function is represented as a weighted sum, but dynamic weight adaptations guide the search towards the direction for which the distance is largest.

The **MDLS principle** adopted in this paper was introduced by Tricoire [16], approximating the Pareto frontier without requiring a priori choices regarding the importance of the objectives considered. An iteration starts from a set of non-comparable solutions, one of which is selected to perform operations. The purpose is to obtain new solutions which are either dominating or non-comparable to the selected one, implying that they should improve the selected solution with respect to at least one objective. Therefore, it is sufficient to perform local search in each direction separately, always restarting from the initially selected solution. Although this paper considers the bi-objective case, any number of objectives can be included from a conceptual point of view [16] by optimizing the selected solution in every relevant direction. Analogous local search operators can be used or different operators can be tailored to each direction, either embedded in a metaheuristic framework or not. After each iteration, new solutions are added to the approximation set and dominated or identical solutions are removed. The fact that each iteration starts from an already efficient solution in this approximation set, rather than from a new initial solution, limits the complexity of the local search phase. Furthermore, MDLS intrinsically delivers a well-spread set of solutions. Even though only one of them will actually be executed, this set of solutions provides insight into the tradeoff between operational costs and service quality, which are both important concerns while solving a DARP. For example, defining a priori weights for both objectives may result in a single solution for which the quality level could considerably be improved by allowing a slight increase in total distance traveled, whereas such a tradeoff would immediately become clear using a bi-objective approach.

## 3 Problem description

This paper adopts the standard **problem characteristics** defined by Cordeau and Laporte [3]. Three types of service-related constraints are included. First, a user is allowed to define a preference time for either his departure or his arrival. A maximum deviation is linked to this preference time, resulting in a time window for either the origin or the destination involved with a user's trip. Second, a maximum user ride time is taken into account, being the maximum time a user can spend aboard the vehicle, which implicitly results in a time window for the other location as well. Third, a service

duration is imposed, being the time a customer needs to get on or off the vehicle. Additionally, all technical constraints related to a pickup and delivery problem [12] should be respected, including load, maximum route duration, pairing and precedence constraints. The objective function minimizes total ride time experienced by all users, which represents a quality dimension, and total distance traveled by all vehicles, being an operational measure.

In a subsequent phase, additional problem characteristics will be incorporated, based on suggestions in quality surveys among customers and experiences of service providers in daily practice. Some relevant extensions are discussed in section 6.

## 4 Implementation of MDLS

An **initial solution set** is obtained using multiple runs of a simple insertion heuristic, employing a weighted-sum objective function. Different weight combinations can be applied to stimulate the creation of a diverse set. In the first run for each weight combination, customers are ordered by the upper bound of their critical time window. In all following runs, they are inserted in random order. At the end, dominated or identical solutions are removed from the resulting initial solution set.

Within the MDLS procedure, local search is guided by a **VNS framework** [7] which includes the same operator types for both objectives. The **relocate** operator repeatedly moves a single customer to the best feasible position in any route. To select the customer to be moved next, a candidate list is constructed. Depending on the objective considered, this list contains all users in descending order of either the excess ride time they experience or the detour a vehicle should make to serve them. The first-ranked customer is the first one for whom a relocate move is investigated, since this might bring about a relatively large improvement of the objective value. If an improving relocation can be found, the candidate list is entirely reconstructed and the procedure restarts. If no improvement can be obtained, the first candidate is removed from the current list and the next-ranked user is considered, etc. The relocate operator terminates as soon as no further improvement can be reached for any of the customers. The candidate list strategy embodies the principle of gradually enlarging neighborhoods, which is often included in VNS approaches, and particularly avoids that computation time is spent on uninteresting candidates. The **exchange** operator repeatedly performs the best feasible exchange of two customers. To select one of both customers to be exchanged next, the candidate list principle is applied. The selected customer can be inserted at the exact position of any other customer, who is in turn inserted at the best position in the route of the originally selected customer. The **natural sequences** operator repeatedly performs the best feasible exchange of two natural node sequences. Such natural sequences are found between two arcs which are traversed by an empty vehicle, implying that they can be exchanged without violating pairing constraints. For this operator, the candidate list contains empty arcs in descending order of length. As a long empty arc in a route indicates that different parts of this route do not correspond well to each other, it might be beneficial to exchange the natural sequence starting or ending at this arc with a natural sequence in any other route.

For each objective separately, an **iteration of MDLS** consists of selecting one solution from the set of non-comparable solutions and successively executing all VNS operators until a local optimum is reached. From the second iteration, the selected solution might already have been optimized with respect to one of both objectives. Therefore, a diversification is applied first, using a **destroy and repair** operator which removes and reinserts a certain percentage of the requests. The principle of Shaw removal [15] ensures temporal and spatial similarities between these requests, thereby increasing the probability that a different feasible solution can be constructed. This new solution is obtained by means of 2-regret insertion [5], which partly remedies the myopic behavior of a classical insertion heuristic by comparing the best and second-best insertion positions for each unassigned request.

Checking the feasibility of a route requires the construction of a time schedule, since time-related constraints (time windows, maximum user ride time and maximum route duration) can only be checked if the start time of the service in each node is known. The presence of maximum user ride time constraints complicates the scheduling phase, since it is not necessarily optimal to depart from a node at the earliest possible time. Most single-objective solution methods invoke a well-known eight-step procedure [3], based on the notion of forward time slack [14]. However, as this procedure does not minimize total user ride time, it is unsuitable for the bi-objective context considered here. Addressing this issue by adapting the forward time slack computation makes the procedure too restrictive in terms of feasibility [13]. Therefore, a new and more efficient heuristic **scheduling procedure** is used, whose fundamental structure focusses on a minimization of total user ride time. This procedure starts from a (possibly infeasible) schedule in which total user ride time equals its lower bound. It gradually establishes feasibility while avoiding unnecessarily large increases in total user ride time. Infeasible routes are discovered in several ways, after which the procedure is aborted prematurely to save computation time. Compared with other heuristic scheduling procedures [13], solutions match the exact time schedule more frequently, the remaining deviations from optimality are smaller and the risk of incorrect infeasibility declarations is strongly reduced.

## 5 Computational tests

The MDLS framework is tested on **benchmark data** [2] containing instances with up to 48 requests and four vehicles, on which the problem characteristics discussed earlier [3] apply. Optimal solutions for these instances are known in the single-objective context of minimizing total distance traveled.

First, the VNS framework within the MDLS procedure is tested separately in this **single-objective** context. Four runs are performed for each instance. Each run is initiated by a single execution of the insertion heuristic, placing the entire weight on the distance objective. In the first run, users are sorted according to their time window, whereas the insertion order is random for all following runs. Next, five MDLS iterations (taking into account only one direction) are applied as described in section 4. Over all problem instances, the best result in four runs is situated at 0.18% from optimality, whereas the optimality gap of the average result in four runs amounts to 0.80%. Comparing the first run with all others, limited differences in solution quality are found. Besides, all operators deliver contributions in the development of the solution, including the destroy-and-repair diversification. Computation times are comparable to other single-objective local-search based metaheuristics [1, 11].

Second, the actual MDLS procedure is tested for the **multi-objective** context presented in this paper. Its approximations of the Pareto frontier are compared to the exact solutions, which are computed by extending the single-objective branch-and-cut algorithm of Braekers et al. [1] using the epsilon constraint technique. To compare solutions in a multi-objective context, different sorting criteria can be used. For the purpose of this test, the two most common indicators are applied, being the hypervolume indicator [18] and the unary epsilon indicator [19].

## 6 Future work

In order to enlarge the applicability of the MDLS metaheuristic presented in this paper, additional real-life characteristics will be added to the current problem context. These may relate to additional constraints faced by service providers in daily practice, as well as to additional objective components of which the usefulness was illustrated by quality surveys. Suggestions by service providers typically

focus on a limited availability of information, such as the fact that dynamic demand or stochastic traffic circumstances should be taken into account to increase the responsiveness and reliability of the system. Other suggestions concern a more sophisticated service design. Apart from heterogeneous users and driver breaks, which are already included in existing multi-objective models, it might be useful to consider driver qualifications or targeted combinations of particular user types. With regard to additional objective components, the MDLS framework is extendable to any number of objectives. They might relate to relevant operational objectives, such as minimizing required fleet size during peak periods, or to quality-oriented goals, such as minimizing waiting time faced by users. Recent surveys on service quality and user satisfaction [8, 9] still reveal an insufficient match between customers' concerns and the problem characteristics taken into account by solution techniques.

## References

- [1] Kris Braekers, An Caris, and Gerrit K. Janssens. Exact and meta-heuristic approach for a general heterogeneous dial-a-ride problem with multiple depots. *Transportation Research Part B: Methodological*, 67:166–186, 2014.
- [2] Jean-François Cordeau. A branch-and-cut algorithm for the dial-a-ride problem. *Operations Research*, 54(3):573–586, 2006.
- [3] Jean-François Cordeau and Gilbert Laporte. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, 37(6):579–594, 2003.
- [4] Jean-François Cordeau and Gilbert Laporte. The dial-a-ride problem: Models and algorithms. *Annals of Operations Research*, 153(1):29–46, 2007.
- [5] Marco Diana and Maged M. Dessouky. A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows. *Transportation Research Part B: Methodological*, 38(6):539–557, 2004.
- [6] Fabien Lehuédé, Renaud Masson, Sophie N. Parragh, Olivier Péton, and Fabien Tricoire. A multi-criteria large neighbourhood search for the transportation of disabled people. *Journal of the Operational Research Society*, 65(7):983–1000, 2013.
- [7] Nenad Mladenović and Pierre Hansen. Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100, 1997.
- [8] Julie Paquette, François Bellavance, Jean-François Cordeau, and Gilbert Laporte. Measuring quality of service in dial-a-ride operations: the case of a Canadian city. *Transportation*, 39(3):539–564, 2012.
- [9] Julie Paquette, Jean-François Cordeau, and Gilbert Laporte. Quality of service in dial-a-ride operations. *Computers & Industrial Engineering*, 56(4):1721–1734, 2009.
- [10] Julie Paquette, Jean-François Cordeau, Gilbert Laporte, and Marta M. B. Pascoal. Combining multicriteria analysis and tabu search for dial-a-ride problems. *Transportation Research Part B: Methodological*, 52:1–16, 2013.
- [11] Sophie N. Parragh. Introducing heterogeneous users and vehicles into models and algorithms for the dial-a-ride problem. *Transportation Research Part C: Emerging Technologies*, 19(5):912–930, 2011.

- [12] Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. Part II: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58(2):81–117, 2008.
- [13] Sophie N. Parragh, Karl F. Doerner, Richard F. Hartl, and Xavier Gandibleux. A heuristic two-phase solution approach for the multi-objective dial-a-ride problem. *Networks*, 54(4):227–242, 2009.
- [14] Martin WP Savelsbergh. The vehicle routing problem with time windows: Minimizing route duration. *ORSA Journal on Computing*, 4(2):146–154, 1992.
- [15] Paul Shaw. A new local search algorithm providing high quality solutions to vehicle routing problems. *APES Group, Dept of Computer Science, University of Strathclyde, Glasgow, Scotland, UK*, 1997.
- [16] Fabien Tricoire. Multi-directional local search. *Computers & Operations Research*, 39(12):3089–3101, 2012.
- [17] Roberto Wolfler Calvo and Alberto Colomi. An effective and fast heuristic for the dial-a-ride problem. *4OR: A Quarterly Journal of Operations Research*, 5(1):61–73, 2007.
- [18] Eckart Zitzler and Lothar Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999.
- [19] Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, 2003.