Computers & Operations Research 56 (2015) 93–109

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Testing local search move operators on the vehicle routing problem with split deliveries and time windows

Marcus E. McNabb*, Jeffery D. Weir, Raymond R. Hill, Shane N. Hall

Air Force Institute of Technology, 2950 Hobson Way, Wright Patterson AFB, OH 45433, USA

ARTICLE INFO

Available online 26 November 2014

Keywords:
Heuristic
Vehicle routing problem
Time windows
Split delivery
Ant colony optimization
Local search

ABSTRACT

The vehicle routing problem (VRP) is an important transportation problem. The literature addresses several extensions of this problem, including variants having delivery time windows associated with customers and variants allowing split deliveries to customers. The problem extension including both of these variations has received less attention in the literature. This research effort sheds further light on this problem. Specifically, this paper analyzes the effects of combinations of local search (LS) move operators commonly used on the VRP and its variants. We find when paired with a MAX-MIN Ant System constructive heuristic, Or-opt or 2-opt* appear to be the ideal LS operators to employ on the VRP with split deliveries and time windows with Or-opt finding higher quality solutions and 2-opt* requiring less run time.

Published by Elsevier Ltd.

1. Introduction

The vehicle routing problem (VRP) is an important transportation problem that seeks an optimal solution for constructing delivery routes given a depot, a fleet of vehicles and a some number of geographically dispersed customers, each having a demand that must be fulfilled. The problem also incorporates characteristics such as travel times and/or distances as well as side constraints such as a maximum vehicle load. This problem is important due to both its widespread application and its complexity in solving. See [1] for a more thorough review of the VRP. The literature addresses several extensions of this problem, including variants having delivery time windows associated with customers (VRPTW) and variants allowing split deliveries to customers (SDVRP). The problem extension including both of these variations has received less attention in the literature. This research sheds further light on this problem, which is important because the addition of these two features more accurately represents important real-world applications of the VRP. Furthermore, the problem and methods used to approach the problem may differ significantly in the presence of these additional characteristics, implying the need for research expressly dedicated to these variants.

Others have explored the effect of splitting loads in further detail. For example, Archetti et al. [2] show empirically the value of the splitting option in a VRP, concluding the splitting option is most effective in problems with a mean customer demand of just over half of the vehicle capacity. Nowak et al. [3] reach a similar conclusion for a pickup and delivery problem with split loads (PDPSL). Nowak et al. [4] also delve into the problem characteristics and empirically show their impact on the solution. However, this research focuses on a problem with both split deliveries and time windows. In the context of the vehicle routing problem with divisible deliveries and pickups (VRPDDP), Nagy et al. [5] claim "one should expect that if there are very tight maximum time constraints applied, then splitting is unlikely to be beneficial, because vehicles will not be filled to capacity anyway." Further research into this conjecture [6] supports this statement, showing the impact of time windows on the solution tends to increase as the mean customer demand decreases relative to the vehicle capacity and the vehicles are generally only filled to about 2/3 capacity in those instances. Logically, this conclusion makes sense because more splits means more deliveries, and in the presence of tight time window constraints, more deliveries puts more stress on the time constraints. Therefore, the inclusion of time windows calls into question the validity of broadly extending results from the PDPSL, VRPDDP, and even the SDVRP directly to the SDVRPTW because the presence of time windows may significantly alter the problem structure and therefore the solution structure. Furthermore, the ratio of customer demand to vehicle capacity in the problem sets used here (see Section 3.2) does not meet the threshold of one half agreed upon by these sources. Therefore, an empirical analysis into the characteristics of the SDVRPTW and their impact on solution quality, as well as a comparison of those results with the work from the SDVRP, PDPSL, and VRPDDP, is warranted and accomplished in a separate article (see [6]). This particular vein of research focuses on the effect of the choice of LS

* Corresponding author. Tel.: +1 937 255 3636x4595.
E-mail address: marcus.mcnabb@us.af.mil (M.E. McNabb).

http://dx.doi.org/10.1016/j.cor.2014.11.007
0305-0548/Published by Elsevier Ltd.

on the solution quality and run time, a factor none of the previous works address.

Specifically, this paper analyzes the effects of combinations of local search (LS) move operators commonly used on the VRP and its variants to empirically determine the combination best suited to generating good solutions for the VRP with split deliveries and time windows (SDVRPTW) within an ant colony optimization (ACO) metaheuristic and is organized as follows. Section 2 presents background on the problem and provides a literature review, Section 3 describes the test problems and experimental design for the computational results presented in Section 4, and, finally, Section 5 concludes with findings and areas for future research.

2. Background

This section will cover the relevant literature for the SDVRPTW with a brief overview on LS operators and the ACO metaheuristic. This section will also discuss these heuristics as they have been applied to the VRP, focusing specifically on applications involving the VRPTW, SDVRP, or SDVRPTW.

2.1. LS for SDVRPTW

Archetti and Speranza [7] offer a concise review of existing work for the SDVRP. They cover both heuristic and exact methods employed thus far, emphasizing the improvements in solutions to various test problems seen when comparing traditional VRP solutions without split deliveries to solutions allowing split deliveries. This research will focus in particular on the applications of LS operators from these research efforts.

Feillet et al. [8] use a branch-and-price algorithm to solve examples of the SDVRPTW exactly. However, like the VRP and many of its variants, the SDVRPTW is NP-hard [9] and exact solutions are difficult to come by, generally requiring extremely long computation times. Frizzell and Giffin [10] first introduce LS to the SDVRPTW, pairing two operators – moving a customer to a new route or swapping customers between routes – with a look-ahead construction heuristic. They employ the LS on problems using grid network distances. Ho and Haugland [11] use a tabu search to tackle the SDVRPTW, employing the following LS operators: Relocate – moves a customer to new route; Relocate-split – splits a customer's load and moves those two loads to new routes; Exchange – trades a pair of customers on separate routes; and 2-opt⁎ – exchanges the last m customers from one route with the last n customers of another route. Campos et al. [12] adapt the Clarke–Wright savings algorithm to the SDVRPTW to develop an initial solution and then use a genetic algorithm to improve this initial solution. Belfiore et al. [9] use scatter search to generate solutions for the SDVRPTW.

Many LS operators are employed in approximating solutions for the VRP and its variants. Some of the most popular or promising operators are now discussed. As seen above, Ho and Haugland [11] successfully utilize four LS operators (Relocate, Relocate-split, Exchange, and 2-opt⁎) on the SDVRPTW. In addition to these operators, one question this research will address is how well LS operators from the VRPTW and SDVRP variants extend to the SDVRPTW. Dror and Trudeau [13], generally regarded as the first to investigate the SDVRP, introduce the 2-split-interchange LS operator, which is also the basis for the Relocate-split operator described above. Aleman et al. [14]introduce a Shift⁎ operator for the SDVRP. The Shift⁎ operator is similar to the Exchange operator described above except it allows for a partial shift of one of the customers. Derigs et al. [15] introduce a series of LS operators specific to the SDVRP, including Combine, Relocate, and another operator similar to the Relocate-Split LS operator; additionally, the authors introduce the concept of combining a split delivery and introducing a new route for this delivery. Braysy and Gendreau [16] detail many LS operators used to generate solutions for the VRPTW, including 2-opt⁎, Or-opt, and Cross Exchange. These three LS operators prove very popular and effective in the VRP literature (see [17–23]). Each of these methods is described in further detail in Section 4. For further details, see [7] for the SDVRP and [1] and [16] for the VRPTW.

2.2. LS performance analysis

None of the LS implementations on the SDVRPTW discussed above make any explicit argument for why a particular LS operator is chosen. None tested the LS operators to show the one (or several) chosen was the best choice for the problem. Rather, LS operators are most likely chosen based on successful implementations on other variants of the VRP.

Others have undertaken the task of comparing the performances of LS operators for several variants of the VRP and related problems, but none have specifically investigated the SDVRPTW. Stutzle [24] investigates the effects of several LS operators on the traveling salesman problem, the quadratic assignment problem, and the flow shop problem when paired with an ACO metaheuristic. Van Breedam [25] analyzes the effectiveness of several LS operators, paired with several different solution construction heuristics, for the VRPTW and the pickup and delivery problem. Braysy and Gendreau [16] further analyze LS operators when applied to the VRPTW. Derigs et al. [15] investigate the effects of LS operators on the SDVRP. However, this literature review revealed no work done to investigate the effects of LS operators when applied to the SDVRPTW.

2.3. Metaheuristics

Testing the performance of the LS operators requires combining these LS operators with a construction heuristic into a metaheuristic. This research effort uses an ACO metaheuristic. This metaheuristic is chosen for two reasons: first, it is successfully implemented on the VRP and several of its variations (see [20–22,26]); and second, it is studied less extensively than other metaheuristics such as tabu search (see [1,27]). The ACO metaheuristic was first introduced by Dorigo [28]. The ACO metaheuristic iteratively constructs a series of solutions [29] where each ant provides an instance of a solution construction. Ants probabilistically add components to their individual solutions until reaching a complete solution. The addition of components is based on heuristic and pheromone information about the problem. In the case of a VRP, the heuristic information consists of the edge costs (e.g., cost or time to transit a commodity over a given edge). The pheromone information is gleaned from previous solutions. More specifically, each edge is initialized with the same amount of pheromone. As a portfolio of solutions is built, a local pheromone update decreases the pheromone on those edges used in building a solution while a global pheromone update deposits additional pheromone onto the "good" edges. In general, a "good" edge is one included in what is deemed a high-quality solution (e.g., "global best" or "iteration best" solution). The local pheromone update encourages exploration of new solutions while the global update encourages exploitation of high-quality solutions.

A literature review of ACO algorithms reveals the application of LS greatly enhances the performance of many ACO implementations [29]. The LS is generally implemented after an ant has constructed a complete solution, at which time the LS attempts to improve this solution. This coupling tends to work well because ACO algorithms perform a rather coarse-grained search meaning a solution is generally amenable to improvement via LS. Meanwhile the primary issue with LS is the generation of a starting solution. Therefore, the combination of these two methods tends to yield