Dynamic Vessel-to-Vessel Routing Using Level-wise Evolutionary Optimization

Yash Vesikar, Julian Blank, Kalyanmoy Deb
Michigan State University
East Lansing, Michigan
{vesikary,blankjul,kdeb}@msu.edu

Markku Kalio
Aalto University
Espoo, Finland
markku.kallio@aalto.fi

Alaleh Maskooki
Univrsity of Turku
Turku, Finland
alaleh.a.maskooki@utu.fi

ABSTRACT

Modern practical optimization problems are too often complex, non-linear, large-dimensional, and sometimes dynamic making gradient-based and convex optimization methods too inefficient. Moreover, most such problems which must be solved for a reasonably approximate solution routinely in every few hours or every day must use a computationally fast algorithm. In this paper, we present a formulation of a dynamic vessel-to-vessel service ship scheduling problem. In a span of several hours, the service ship must visit as many moving vessels as possible and complete the trip in as small a travel time as possible. Thus, the problem is bi-objective in nature and involves a time-dependent traveling salesman problem. We develop a level-wise customized evolutionary algorithm to find multiple trade-off solutions in a generative manner. Compared to a mixed-integer programming (MIP) algorithm, we demonstrate that our customized evolutionary algorithm achieves similar quality schedules in a fraction of the time required by the MIP solver. We are currently developing an interactive decision support tool based on our proposed method for finding multiple trade-off schedules simultaneously and selecting a single preferred one.

CCS CONCEPTS

• **Mathematics of computing → Graph algorithms**.

KEYWORDS

Multi-objective Optimization, Graph Search, Real-World Application, Combinatorial Optimization

ACM Reference Format:
Yash Vesikar, Julian Blank, Kalyanmoy Deb, Markku Kalio, and Alaleh Maskooki. 2020. Dynamic Vessel-to-Vessel Routing Using Level-wise Evolutionary Optimization. In *Genetic and Evolutionary Computation Conference Companion (GECCO '20 Companion), July 8–12, 2020, Cancún, Mexico.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3377929.3389910

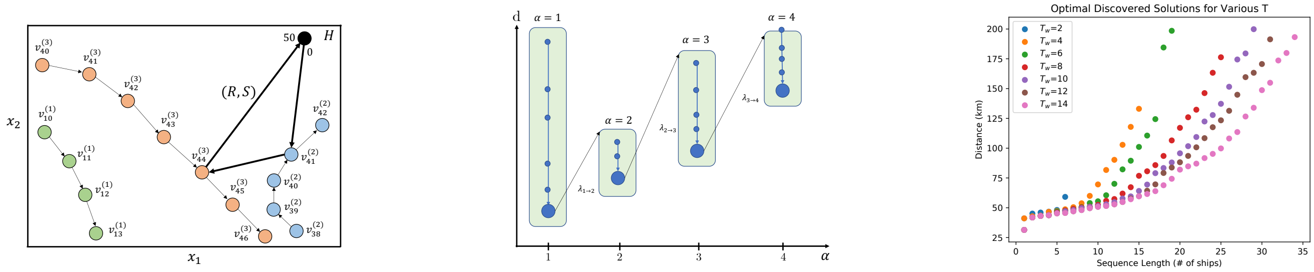
1 INTRODUCTION

A starting point to tackle real-world optimization tasks is to simplify them as tractable optimization problems, so that a standard provable algorithm can be applied. However, the obtained solutions to the simplified problems may not represent the true properties of the optimal solution to the original problem. There is a need to keep the original complexity of the problem and devise computationally tractable and customized algorithms which can produce near-optimal solutions quickly. In this paper, we investigate the dynamic vessel-to-vessel routing problem (DV2VRP), where a route for a service ship, starting from and ending at a harbor, is to be found by minimizing the traveling time and maximizing the number of ships visited. DV2RP can be viewed as a dynamic and bi-objective TSP problem in which location of nodes change with time and also a subset of nodes is allowed in a tour, making the problem extremely challenging to solve. We propose a level-wise genetic algorithm (LW-GA) with customized mutation and crossover operators to find near-optimal solutions for DV2VRP. The time dependent dynamics are addressed by forming multiple levels of optimization, where the route-level problem searches for a tour and the schedule-level problem searches for a time schedule of visiting vessels dictated at the route-level.

ACM ISBN 978-1-4503-7127-8/20/07.
https://doi.org/10.1145/3377929.3389910

1.1 Problem Formulation

DV2VRP is a bi-objective optimization problem where a service ship needs to visit the largest number of vessels in the shortest possible time. The service ship starts at a harbor H and needs to return back to harbor after visiting a number of vessels within a pre-specified time window T_w. The two objectives of the task are: (i) maximize the number of vessels visited (α) by the service ship, and (ii) minimize the traveling time d ($< T_w$). The problem defines N vessels in total, where k-th vessel $\boldsymbol{v}^{(k)}$ are at different locations at different time slots. The notation $v_j^{(k)}$ denotes k-th vessel location at the j-th time step (see Figure 1a). Note that the locations are based on each vessel's route which can be assumed to be pre-determined and that $v_t^{(k)}$ is only available exactly at time t, not before or after. In total, there exist $\sum_{\alpha=1}^{N}\left(\binom{N}{\alpha}\alpha!\right)$ possible routes. The figure shows a two-vessel ($\alpha = 2$) route in a three-vessel ($N = 3$) problem: Harbor to Vessel-2 to Vessel-3 and back to Harbor. Also, for a given route involving α vessels, a schedule involving time slot for visiting each vessel in a chronological manner as dictated by the route will finally dictate the overall travel time. Note that many of the routes will not be feasible, as some vessels may have sailed out of reach at the time slot of consideration and many schedules are not practically feasible due to time continuity violations. To determine the feasibility, we denote available time slots of k-th vessel as $\Omega(v^{(k)})$. For the example, the availability of vessel $\boldsymbol{v}^{(1)}$ in the figure is given by $\Omega(\boldsymbol{v}^{(1)}) = (10, \ldots, 13)$. The specific schedule marked in the figure indicates that the service ship starts from harbor at zero-th time slot, reaches Vessel 2 at 41-st time slot, then goes to Vessel 3 at 44-th time slot, and finally returns



(a) Illustration of a DV2VRP with the route $R = (H, 2, 3, H)$ and the schedule $S = (0, 41, 44, 50)$.

(b) Visual walkthrough of the optimization procedure in the upper level.

(c) Discovered Pareto-Optimal solutions for different time horizons T_w.

to harbor at 50-th time slot – all are within their Ω-vectors. Thus, a solution to DV2VRP with α vessels consists of two entities: (i) a route $R = (H, P_1, \ldots, P_\alpha, H)$, in which P_i denotes the identity of i-th vessel in the route, and (ii) a schedule $S = (0, t_{P_1}, \ldots, t_{P_\alpha}, t_H)$, in which t_{P_i} denotes the time slot for vessel P_i. The traveling time between i-th vessel at p-th time slot and j-th vessel at q-th time slot is denoted by $c(v_p^{(i)}, v_q^{(j)})$. Then, the total travel time for a α-vessel route R with a schedule S is given as

$$d(R, S) = \sum_{k=0}^{\alpha} c\left(v_{S_k}^{(R_k)}, v_{S_{k+1}}^{(R_{k+1})}\right). \qquad (1)$$

The harbor $v^{(H)}$ is available at any time. By representing a solution as (R, S), a formulation of DV2VRP is given below:

$$\begin{aligned} \min_{(R,S)} \quad & (-\alpha(R, S),\ d(R, S)), \\ \text{subject to} \quad & \text{(i) } d(R, S) \le T_w, \\ & \text{(ii) } S_k \in \Omega(v^{(R_k)}), \quad k = 1, \ldots, \alpha, \\ & \text{(iii) } S_k + c\left(v_{S_k}^{(R_k)}, v_{S_{k+1}}^{(R_{k+1})}\right) \le S_{k+1},\ k = 1, \ldots, \alpha. \end{aligned} \qquad (2)$$

The constraints ensure that (i) the service ship returns to the harbor before the time horizon limit exceeds, (ii) the schedule contains only valid time slots where a vessel is available and (iii) each transition between vessels (or harbor and vessel) is feasible and does not violate the time constraint.

2 METHODOLOGY

We propose LW-GA a multi-level algorithm to optimize DV2VRP. The first level reduces the bi-objective problem to a single-objective problem by using by fixing α. The second level problem searches for a route R given α and the third level problem develops a schedule that minimizes the traveling time $d(R, S)$, given a route R of size α.

α-level Problem

The overall procedure starts with $\alpha = 1$ (first sub-problem) and continues with increasing α, until no feasible solution can be found (see Figure 1b). After having found solution(s) that minimize the travel time $d(\alpha)$ given α, a transition function $\lambda_{\alpha \to \alpha+1}$ transfers the obtained solutions to the next sub-problem $\alpha + 1$ by increasing the length of each route R and schedule S by one. We have used a heuristic based optimization for $\lambda_{\alpha \to \alpha+1}$ to ensure a fast but effective transition.

Route-level Problem for a Given α

The route-level optimization problem is responsible for finding a vessel route $R = (H, P_1, \ldots, P_\alpha, H)$ using *exactly* α ships for the schedule-level optimization to evaluate. A route always starts from the harbor and ends at the harbor visiting each vessel at most once. Such a schedule provides information about feasibility and minimum travel time $d^*(R)$ for a route which is used as the constraint and objective value at the route-level respectively. We use a genetic algorithm with customized heuristic-based crossover and mutation operators which ensure the offspring being feasible. Ensuring feasibility is crucial because each function evaluation requires choosing the exact time schedule S of visiting each vessel which is solved at the schedule-level. In our experiments we have investigated a DV2VRP with $N = 63$ vessels where a permutation-based customized GA converged in an average run in less than 100 generations with a population size of 20.

Schedule-Level Problem for a Given Route of size α

At the lowest level, a route R containing α vessels is supplied, and a schedule S that minimizes the traveling time (Equation 1) needs to be found. We have implemented a dynamic programming approach which considers each transition in a route R from vessel $\boldsymbol{v}^{(R_k)}$ to $\boldsymbol{v}^{(R_{k+1})}$. Let us denote the minimum traveling time from the harbor to vessel $\boldsymbol{v}_t^{(R_k)}$ by $d^*(\boldsymbol{v}_t^{(R_k)})$. Then, the minimum traveling time to $\boldsymbol{v}_t^{(R_{k+1})}$ is given by:

$$d^*(\boldsymbol{v}_t^{(R_{k+1})}) = \min_{q \in \Omega(\boldsymbol{v}^{(R_k)})} d^*(\boldsymbol{v}_q^{(R_k)}) + c(\boldsymbol{v}_q^{(R_k)}, \boldsymbol{v}_t^{(R_{k+1})}). \qquad (3)$$

Initially, $d^*(\boldsymbol{v}_t^{(H)}) = 0$ since the service ship does not leave the harbor yet. Then using this sub-optimality criterion for each transition, the minimum path length is calculated.

3 RESULTS AND DISCUSSIONS

Figure 1c shows the discovered Pareto-optimal solutions using the proposed LW-GA for various time horizons T_w on problem based on the real-world data obtained from a shipping company. We have made use of the multi-objective evolutionary optimization framework pymoo [1] and developed customized evolutionary operators for the problem. Results from the initial studies of LW-GA are promising and are comparable with the known Pareto-optimal values presented in the original work [2]. A standard mixed-integer programming solver requires 60 hours of computational time to solve the $T_w = 10$ hour problem. In contrast, our approach requires only 2.44 minutes to solve the same problem. More results will be communicated in a later publication.

REFERENCES

[1] Julian Blank and Kalyanmoy Deb. 2020. pymoo: Multi-objective Optimization in Python. (2020). arXiv:cs.NE/2002.04504
[2] A. Maskooki and Y. Nikulin. 2018. *Multiobjective Efficient Routing In a Dynamic Network*. Technical Report 1198, Turku Center for Comp. Sc. (TUCS), Finland.