A Two-Stage Set Partioning Approach for the EURO Meets NeurIPS 2022 Vehicle Routing Competition

Florentin D. Hildebrandt
Chair of Management Science
Otto-von-Guericke-Universität
florentin.hildebrandt@ovgu.de

Roberto Roberti
Department of Information Engineering
Università degli Studi di Padova
roberto.roberti@unipd.it

Barrett W. Thomas
Tippie College of Business
University of Iowa
barrett-thomas@uiowa.edu

Marlin W. Ulmer
Chair of Management Science
Otto-von-Guericke-Universität Magdeburg
marlin.ulmer@ovgu.de

Abstract

This document describes ORberto Hood and the Barrymen's submission to the EURO Meets NeurIPS 2022 Vehicle Routing Competition. [1]

1 Foreword

In this report, we present our solution method for the *EURO Meets NeurIPS 2022 Vehicle Routing Competition*. We compete under the name *ORberto Hood and the Barrymen*. The competition consists of a static VRP and a dynamic VRP. For the full description of both problems, we refer to the competition website. We address only the dynamic VRP in this report because we did not adapt (except for hyperparameter-tuning) the solver for the static VRP that is provided to all participants of the competition.

2 Solving the dynamic VRP

A solution in each decision step of the dynamic VRP is derived in two stages. In the first stage, the set of currently known customers is partitioned into the set of customers to serve now and the set of customers to serve in the future. In the second stage, routes are constructed for the set of customers that are served now. The second stage is performed by the routing heuristic that was provided to all participants of the challenge. Hence, we focus our report on the first stage of the problem, i.e., the partitioning of the customers.

At a given time epoch t, we are given a set of customers C_t that have been disclosed but have not been served yet. Some of these customers must be served in time epoch t. Other customer must be served in the next time epochs. The remaining customers $\overline{C}_t$ can be served either in time epoch t or future time epochs. We must decide which customers in $\overline{C}_t$ to serve in the current time epoch and which customer to postpone to the next time epochs. We require perfect information on future customer requests in order to compute the exact cost of postponing customers in $\overline{C}_t$ and, subsequently, to derive an optimal decision in the set partitioning problem. As this information is not available we must approximate the unknown, stochastic future customer requests. For that purpose, we rely on approximate dynamic programming (see [2]). Approximate dynamic programming is a diverse toolbox to tackle sequential decision problems. For the given dynamic VRP, we choose a *stochastic lookahead* from the ADP toolbox. This approach boils down to extending the set of customers C_t by sampling demand scenarios $\mathcal{S}_t$ to approximate future demand. A stochastic lookahead is

EURO Meets NeurIPS 2022 Vehicle Routing Competition Workshop.

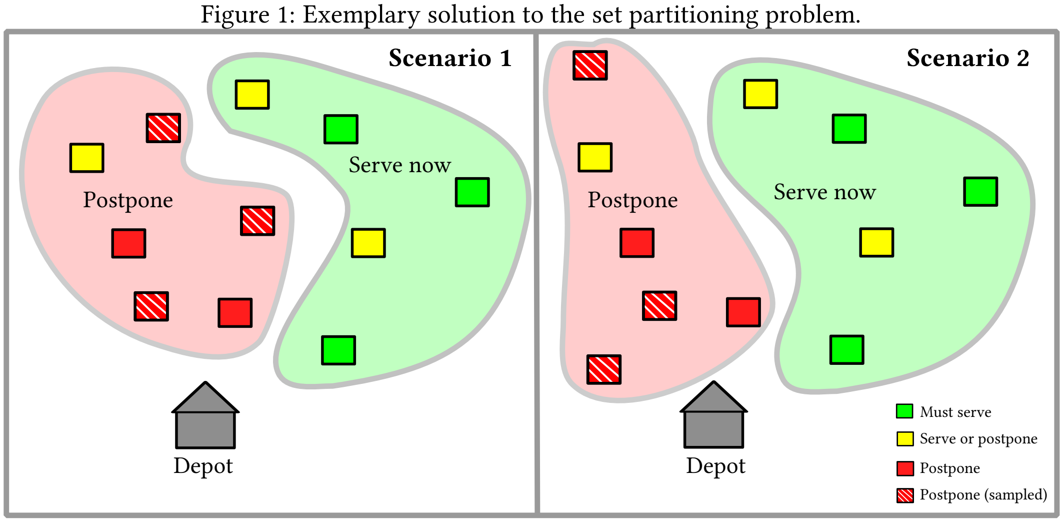
attractive for the given problem because we may sample directly from the true distribution of customer requests (including their time of request, time windows, service times, demand, etc.) and because the scenarios easily integrate into our set partitioning problem formulation. Note that in contrast to the actually revealed customers C_t, the sampled customers C_{tj} of each scenario $j \in \mathcal{S}_t$ are additionally characterized by their release times.

Using the sampled scenarios $\mathcal{S}_t$ to refine our approximation of the marginal cost of serving versus postponing each customer in $\overline{C}_t$ remains a difficult task. Essentially, it requires the integration of a VRP with time-windows, capacity constraints, and release times into the set partitioning problem. In the set partitioning problem, all scenarios S_t are jointly considered as we minimize the routing costs of the customers served now and the mean cost over all scenarios given by the routing cost of the postponed customers and the sampled customers C_{tj}. In other words, postponed customers must be served once for each scenario while customer served now must only be served once for the current state. As a direct consequence, a solution to the set partitioning problem must fulfill the following properties:

1. All *serve now* customers must be served now.
2. All *postpone* customers must be postponed and served in each scenario.
3. Each *serve or postpone* customer must either be served now or must be postponed (and served) in all scenarios (the customer cannot be postponed only for a subset of scenarios).
4. Each sampled customer must only be served in the customer's respective scenario.

We illustrate a exemplary solution to the set partitioning problem in Figure 1.

Figure 1: Exemplary solution to the set partitioning problem.



The figure depicts two scenarios. In each scenario, we are given the same set of *must serve* customers (green squares), *serve or postpone* customers (yellow squares), and *postpone* customers (red squares). These are the customers that have been revealed at the start of the decision period t. We are further given sampled *postpone* customers (red cross-hatched squares). These customers are sampled in each scenario and, therefore, differ in each scenario. An exemplary solution to the set partition problem is given by deciding which customers to serve now (green area) and which to postpone (red area).

As mentioned before, partitioning the customers entails solving the integrated VRP to estimate the marginal costs of serving now versus postponing. By solving this VRP, the tightest bound can be achieved by considering routes that (i) are elementary, (ii) fulfill capacity constraints, (iii) satisfy time window constraints, and (iv) adhere to the release times of sampled customers. As such a VRP features an exponential number of variables, we rely on column generation. Further, pricing out feasible routes is NP-hard and too expensive from a computational standpoint. However, we do not necessarily require a feasible solution to the VRP; we only require a reasonable approximation of the marginal costs of serving versus postponing customers in $\overline{C}_t$. Thus, we relax the requirement that routes must be elementary and generate (q,t)-routes (see [3]) – which can be done in pseudo-polynomial time (i.e., $O(Q \cdot (\ell_0 - e_0) \cdot n^2)$).

All our previous assumptions result in the following mixed-integer linear programming formulation of the set partitioning problem. In the formulation, we introduce the following notation:

- $\mathcal{R}_c$: set of feasible routes to deploy in the current time epoch
- $\mathcal{R}_{f,j}$: set of feasible routes to deploy in future time epochs and assuming scenario $j \in \mathcal{S}_t$
- $c_r \in \mathbb{R}_+$: cost of route $r \in \mathcal{R}_c \cup \left(\bigcup_{j \in \mathcal{S}_t} \mathcal{R}_{f,j} \right)$
- $a_{ir} \in \mathbb{N}$: integer coefficient indicating the number of times customer $i \in C_t$ is served in route $r \in \mathcal{R}_c \cup \left(\bigcup_{j \in \mathcal{S}_t} \mathcal{R}_{f,j} \right)$
- $x_r \in \{0,1\}$: binary variable indicating if route $r \in \mathcal{R}_c$ is deployed in the current time epoch
- $y_{rj} \in \{0,1\}$: binary variable indicating if route $r \in \mathcal{R}_{f,j}$ is deployed in future time epochs and assuming scenario $j \in \mathcal{S}$

The mixed-integer linear programming model is given by:

$$\min \sum_{r \in \mathcal{R}_c} c_r x_r + \frac{1}{|\mathcal{S}_t|} \sum_{j \in \mathcal{S}_t} \sum_{r \in \mathcal{R}_{f,j}} c_r y_{rj} \tag{1a}$$

$$\text{s.t.} \sum_{r \in \mathcal{R}_c} a_{ir} x_r + \sum_{r \in \mathcal{R}_{f,j}} a_{ir} y_{rj} = 1 \qquad \forall j \in \mathcal{S}_t, \forall i \in C_t \cup C_{tj} \tag{1b}$$

$$x_r \in \{0,1\} \qquad \forall r \in \mathcal{R}_c \tag{1c}$$

$$y_{rj} \in \{0,1\} \qquad \forall j \in \mathcal{S}_t, \forall r \in \mathcal{R}_{f,j} \tag{1d}$$

This formulation aims to provide a valid dual (i.e., lower) bound to the cost of serving all customers of the set C_t while deciding upon the partitioning of the customers between the current time epoch and future time epochs. A solution to the set partitioning problem, i.e., which customers to serve now, is obtained by

$$C^* := \{c \in C_t \mid \exists r \in \mathcal{R}_c \text{ s.t. } x_r = 1 \text{ and } c \in r\}. \tag{2}$$

Acknowledgments and Disclosure of Funding

Florentin Hildebrandt's research is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), project 413322447. Marlin Ulmer's work is funded by the DFG Emmy Noether Programme, project 444657906. We gratefully acknowledge their support.

References

[1] W. Kool, L. Bliek, D. Numeroso, R. Reijnen, R. R. Afshar, Y. Zhang, T. Catshoek, K. Tierney, E. Uchoa, T. Vidal, and J. Gromicho, "The EURO meets NeurIPS 2022 vehicle routing competition," 2022.

[2] W. B. Powell, H. P. Simao, and B. Bouzaiene-Ayari, "Approximate dynamic programming in transportation and logistics: a unified framework," *EURO Journal on Transportation and Logistics*, vol. 1, no. 3, pp. 237–284, 2012.

[3] M. Desrochers, J. Desrosiers, and M. Solomon, "A new optimization algorithm for the vehicle routing problem with time windows," *Operations research*, vol. 40, no. 2, pp. 342–354, 1992.