WORKING PAPER SERIES

Dynamic Priority Rules for Combining On-Demand Passenger Transportation and Transportation of Goods

Alexander Bosse/Marlin W. Ulmer/Emanuele Manni/Dirk C. Mattfeld

Working Paper No. 6/2022

OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

FACULTY OF ECONOMICS
AND MANAGEMENT

Impressum (§ 5 TMG)
Herausgeber:
Otto-von-Guericke-Universität Magdeburg
Fakultät für Wirtschaftswissenschaft
Der Dekan

Verantwortlich für diese Ausgabe:
A. Bosse, M. W. Ulmer, E. Manni, D. C. Mattfeld
Otto-von-Guericke-Universität Magdeburg
Fakultät für Wirtschaftswissenschaft
Postfach 4120
39016 Magdeburg
Germany

http://www.fww.ovgu.de/femm

Bezug über den Herausgeber
ISSN 1615-4274

Dynamic Priority Rules for Combining On-Demand Passenger Transportation and Transportation of Goods

Alexander Bosse, Marlin W. Ulmer, Emanuele Manni, Dirk C. Mattfeld

Abstract

Urban on-demand transportation services are booming, in both passenger transportation and the transportation of goods. The types of service differ in timeliness and compensation and, until now, providers operate larger fleets separately for each type of service. While this may ensure sufficient resources for lucrative passenger transportation, the separation also leaves consolidation potentials untapped. In this paper, we propose combining both services in an anticipatory way that ensures high passenger service rates while simultaneously transporting a large number of goods. To this end, we introduce a dynamic priority policy that uses a time-dependent percentage of vehicles mainly to serve passengers. To find effective time-dependent parametrizations given a limited number of runtime-expensive simulations, we apply Bayesian Optimization. We show that our anticipatory policy increases revenue and service rates significantly while a myopic combination of service may actually lead to inferior performance compared to using two separate fleets.

Keywords: Routing, Stochastic dynamic vehicle routing, Ride-hailing, Instant delivery, Bayesian Optimization

1 Introduction

The demand for urban mobility and transportation services is constantly increasing. Every day, customers spontaneously demand services such as passenger transportation (e.g., ride-hailing) or the transportation of goods (e.g., instant delivery or courier services). Companies often focus on either mobility or transportation demand, for example, Lyft or MOIA (mobility) and Amazon

Prime Now or GoPuff (delivery of goods). When serving both types of demand, they usually dispatch different fleets, one for each type of service (e.g. UberPool and UberEats).

Both types of service occur at the same time in the same areas. Thus, by focusing on only one type of service or by splitting the fleet, companies miss several consolidation opportunities. For example, a mobility service vehicle could perform transportation of goods nearby mobility demand or at times when mobility demand is relatively small. Serving passengers and transporting goods by one combined fleet comes with additional challenges. Mobility and transportation services differ in both revenue and service requirements. While passenger transportation often provides comparably high revenue, it requires fast service. For the transportation of goods, the revenue is relatively small, but the available time to fulfill the demand may be longer, for example, 2 hours for Amazon Prime Now delivery. Thus, in a combined setting, the question arises how to use the fleet effectively. Focusing only on mobility demand may lead to many missed opportunities for transporting goods and consolidation while satisfying every transportation demand at all cost may consume resources required for later mobility demand. Utilizing the fleet effectively is therefore challenging, especially when the ratio of mobility and transportation requests varies over time.

The resulting optimization problem is a stochastic dynamic pickup and delivery problem with heterogeneous customer requests differing in deadlines and revenue. Over the course of the day, customers request passenger transportation or the transportation of goods. Both types of requests require the timely pickup at a location and the drop off or delivery at another location in the city. We assume the time allowed to satisfy a request depends on the travel time between pickup and delivery, plus some additional time which is shorter for passenger transportation. The revenue per request also depends on the type of service and is significantly higher for passenger transportation. Whenever a customer requests service, decisions are made about if and how the service can be fulfilled by the fleet. If the service is declined, the company loses the corresponding revenue and creates an unhappy customer. Thus, we set the goal to minimizing the lost revenue (later, we show that this also reduces the percentage of unhappy customers).

Deriving effective decision for this problem is challenging for several reasons. First, decisions are made without knowing future demand for both passenger transportation and transportation of goods. Second, decisions impact the fleet's potential for future services. Third, complex assignment and routing decisions need to made in real time for problems with large fleets and many customers. Thus, we propose an intuitive, global strategy. During the day, a part of the fleet serves lucrative passenger requests with priority, i.e. goods are only transported by these vehicles if they are "along the way". Getting the priorities right is challenging as they should capture the current and future demand ratio of passengers and goods as well as consolidation effects. Further, as the demand-ratio varies over time, fixed priority rates are insufficient, a dynamic variation over time is required. Therefore, we propose time-dependent priority rates. A priority policy can therefore be represented by a priority rate function over the time horizon. In our experiments, we compare Fourier-series and Taylor-series (i.e. polynomial) functions as both are often used to approximate more complex functions. For finding the right parametrization of the respective functions, we turn to Bayesian Optimization (BO, Frazier 2018). For a given function space, BO searches the parametrization-space by carefully balancing exploration and exploitation. Our experiments show that combining passenger and good transportation *can* be very beneficial if a fitting strategy is applied. Furhter, priority rules are an intuitive mean to reduce lost revenue significantly. Furthermore, the number of unhappy customers can also be reduced. Making priorities time-dependent is very beneficial, especially when the demand-ratio of passenger and good-requests varies. Finally, relying of Fourier-functions for representing priority rates is advantageous compared to polynomials, likely, because they are restricted and can capture less “clean” developments.

Our paper makes the following contributions: Our work is among the first that considers centralized anticipatory optimization for combined on-demand services of passenger transportation and transportation of goods. We propose an effective and intuitive strategy that allows instant decision making for larger fleets. Our work is the first in dynamic vehicle routing that determines a time-dependent policy-parametrization via Fourier-series functions and Bayesian Optimization. We conduct extensive computational experiments to analyze the value and effect of priority rules for combined services as well as the functionality of BO.

The structure of this article is as follows. In Section 2, we give an overview on the related literature. In Section 3, we present the problem statement and provide a mathematical model. In Section 4, we explain our solution approach. In Section 5, we describe a computational study that was conducted and analyse the results to give managerial implications. We conclude our article in Section 6.

2 Literature Review

In this section, we give an overview of related literature. Our work considers transportation of goods and passengers in a stochastic dynamic setting with a large fleet. We will first review literature on combined transportation of passengers and goods. Then, we will provide an overview on anticipatory policies in stochastic dynamic vehicle routing, especially in combination with fast large-scale optimization.

2.1 Combined Passenger Transport and Transportation of Goods

Work on both dynamic passenger and good transportation services is comparably large, we refer to Soeffker et al. (2022) for a recent survey on dynamic transportation of goods and Tafreshian et al. (2020) for a survey on dynamic passenger transportation. However, work on combining them is

limited. Further, the vast majority of work is done in a static deterministic setting, i.e., all information is known in advance. The majority of this work considers parcel or freight transportation via public transportation systems. We refer to Hörsting and Cleophas (2021) and Elbert and Rentschler (2021) for recent overviews. Most of the considered problems in this field are fully deterministic, only a few consider uncertainty (Ghilas et al. 2016, Mourad et al. 2019, 2021).
Besides using public transport, there is also an increasing amount of work that considers stochastic dynamic combinations of parcel delivery with mobility on demand (see Beirigo et al. 2018 and Fehn et al. 2022 for recent overviews).
Li et al. (2014) consider a ride-sharing problem with freight transportation. They model the static problem and simulate it in a dynamic environment to analyze the impact of different parametrizations. Chen et al. (2017) also consider combined transport of passengers and packages with taxis. They apply a rolling-horizon simulation, however, they prohibit package delivery during rush hours. We show that our policies can combine services even at times when passenger demand is relatively high, achieving high service rates for both services. Chen et al. (2020) extend the work by Chen et al. (2017) by considering multi-hop package delivery via taxis. Manchella et al. also present work on ride-pooling for passengers and multi-hop transfers for goods. They present an agent-based simulation and a reinforcement learning algorithm for repositioning idling vehicles. Thus, while they anticipate future demand in general, they do not consider different types of demand in detail. We note that their repositioning approach may likely complement our work.
Schlenther et al. (2020) present a simulation approach where both passengers and parcels are transported by the same fleet. They propose to not serve all feasible requests but reject requests that require longer travel of the corresponding vehicle. We test a similar concept with our cost-benefit benchmark **CB**. We show that while this leads to improved revenue compared to myopic decision making, it is doing so by transporting only a very limited number of parcels and solely focusing on lucrative passenger transportation. Romano Alho et al. (2021) analyze joint transportation of passengers and parcels with mobility-on-demand vehicles. They provide an agent-based simulation to analyze different problem parameters as well as assignment and repositioning strategies. Meinhardt et al. (2022) considers joint transportation of passengers and freight with autonomous vehicles. Passengers have a higher priority and cannot be served together with parcels. They provide an agent-based simulation to analyze the value of combined services. Finally, in Fehn et al. (2022), combined delivery of parcels and passengers is evaluated via agent-based simulation. The authors show that combining both services is superior to service by two individual fleets.
While the aforementioned papers consider joint transportation, none of the work considers centralized anticipatory optimization as we propose in our work. In many cases, no central decisions are made considering the entire fleet setup, but decisions are made agent-based on individual vehicle level. In our work, we propose a global strategy considering and orchestrating the entire fleet.

Furthermore, for nearly all mentioned papers, decision making is based on reoptimization, i.e., potential future developments are not considered in the assignment and routing of demand. We use a similar strategy called **Myopic** as one of our benchmark policies. In our case study, we show that such a myopic strategy may not always be advantageous compared to individual fleets. That means that in some cases, it might be advantageous to keep the services separate, unless a anticipatory policy is used.

2.2 Stochastic Dynamic Vehicle Routing

Research on stochastic dynamic vehicle routing problems has be receiving increasing attention recently, see Soeffker et al. (2022) for a recent overview.

The most related paper compared to our work is the work by Ghiani et al. (2022). In their work, they analyze courier services with heterogeneous demand classes. Some packages are urgent and have to be delivered fast, others are allowed to take longer. Delayed delivery leads to different penalties dependent on the service class. The goal is to minimize the overall penalty while service all customers. The authors propose reserving a percentage of vehicles exclusively for urgent delivery. They adapt the percentage based on observed demand. The results show that reserving vehicles will improve service for the valuable first class on the expense of the service for lower classes. Our work differs from Ghiani et al. (2022) in problem, model, and methodology. We consider combined transport of passengers and goods with hard time constraints, aiming for a high revenue. Further while our method follows a related idea, serving passengers with priority, the priority fleet can be used to transport goods as well. This leads to improved services for both passenger and good transportation. Finally, our policy aims on longer term anticipation since decisions now impact the state several hours into the future. Thus, instead of focusing on realized demand, we anticipate the demand over the entire day. Our experiments show the benefits of this procedure compared to a more local view of Ghiani et al. (2022) which we adapt in our benchmark policy **TD(CA)**.

Besides the work by Ghiani et al. (2022), several other papers propose anticipatory methods that take decisions with respect to future developments. Some papers transfer analytical considerations into algorithms, e.g., via waiting strategies (Mitrović-Minić et al. 2004, Branke et al. 2005, Thomas 2007), by proactive repositioning strategies (Sheridan et al. 2013), by threshold-policies (Ulmer and Streng 2019, Ulmer et al. 2022), or by adding safety buffers (Ulmer et al. 2021) and penalty terms (Riley et al. 2020). A larger number of papers propose data-driven methods, for example, sampling future developments and incorporating them into decision making (e.g., Bent and Van Hentenryck 2004, Ghiani et al. 2009, Ferrucci et al. 2013, Schilde et al. 2014, Voccia et al. 2019), or simulating the future to evaluate current decisions (e.g. Secomandi 2001, Goodson et al. 2013, Ulmer et al. 2019, Brinkmann et al. 2019). Recently, there has also been an increase in reinforcement learning approaches that iteratively learn the values of different decisions (e.g. Ulmer et al. 2018a,b, Kullman et al. 2021, Al-Kanj et al. 2020, Chen et al. 2022).

The latter two approaches have in common that they are often applied to smaller or simplified problems, either by only considering a few vehicles, decomposing the decision making by focusing on individual vehicles or by condensing the problems to dynamic assignment or resource allocation problems without explicitly routing vehicles (Hildebrandt et al. 2021).

In our problem, real-time decision making for a larger fleet is required and decisions are made about complex routing with pickup and delivery and time constraints. Therefore, we propose a global strategy, focusing on the assignment while considering the entire fleet. We note though that our method is complementary to other, more detailed methods, e.g., for slotting or pricing for customer requests or for routing and repositioning of individual vehicles.

3 Problem Statement

In this section, we first give a general description of the problem at hand. We then propose a formulation of the problem as sequential decision process and explain the different components of the model. Finally, we give a small example to better understand the problem characteristics and model.

3.1 Problem Overview

The problem can be modeled as a stochastic dynamic pickup and delivery problem with heterogeneous services. Over a time horizon, a fleet of capacitated vehicles serves passenger and freight transportation requests. The requests are unknown until the time of request. Each request has an origin, a destination, a deadline for the pickup and the delivery, and a revenue. The deadlines and revenue depend on the type of request and the travel time between origin and destination. For passenger transportation requests, the deadlines are closer to the time of the request and the revenue is larger compared to a freight transportation request. Whenever a new request arises, the provider decides about if service can be offered and how to integrate the request in the routing currently planned for the vehicles. We assume that vehicles cannot be diverted from their next stop in the current route, but routes can only change for later stops. All vehicles start at the same location, the depot. Vehicles do not have to return to the depot after fulfilling the last request of their current route, but wait for new requests or the end of the working day instead. The objective is to minimize the lost revenue in case customer requests cannot be satisfied.

3.2 Mathematical Model

In the following, we describe the problem and decision process as a sequential decision process (Powell 2019). A sequential decision process consists of five key components: decision epochs, states, decisions with reward, stochastic information and a transition function.

3.2.1 Decision Epochs

Decision points refer to points in time in which the decision maker has to decide what actions should be performed next based on the information that is known so far. The time span between two decision points is called decision epoch. In our case, a decision is made whenever a new request arises. We define $K = \{0, 1, \ldots, k_{\max}\}$ as the set of all possible decision points where $k_{\max}$ is the last decision point. Further, each decision point $k \in K$ is associated with a system time $t_k \in T = \{0, 1, \ldots, t_{\max}\}$ where T is the set of all possible system times and $t_{\max}$ is the length of the time horizon.

3.2.2 States

For every decision point, the system is in a certain state. The state in decision point k is called $s_k \in S$ where S is the set of all possible system states. A system state contains information about the current system time t_k. Further it contains information about the next position of the vehicles and when they will be there and their currently planned routes, the set of requests still to serve, whether they are already on a vehicle or not, and their corresponding deadlines. Further, a state contains a new request. The information of vehicles and already accepted requests can be encapsulated in the vehicles' planned routes. We denote the set of routes in state s_k as $r_k = \{r_{k,1}, \ldots, r_{k,n}\}$ where $r_{k,i}$ is the route of vehicle $i \in \{1, \ldots, n\}$ at decision point k. A route $r_{k,i}$ is defined as a sequence of pickup and delivery locations that the vehicle is planned to visit, associated with the arrival time at each location, and in case of delivery, the latest time the delivery needs to take place. For the purpose of presentation, we omit full notation of the routes. For the full model, we refer to Section 6 in Ulmer et al. (2020).
Finally, a state contains a new customer request c_k associated with the time of the request t_k as well as information on the type (either passenger or good), the locations of pickup and delivery, and the revenue R_k. In summary, a system state can be defined as a tuple $s_k = (t_k, r_k, c_k)$.

3.2.3 Decisions

Whenever a new request arises the decision maker has to make decision $x_k \in X(s_k)$ where $X(s_k)$ is the set of all possible actions in system state s_k. In our case, the decision contains two parts,

$x_k = (\alpha_k^x, r_k^x)$. The first part indicated by $\alpha_k^x \in \{0, 1\}$ is the decision of the request is acceptance for service ($\alpha_k^x = 1$) or not ($\alpha_k^x = 0$). The second part r_k^x is about the update of the routing. A decision is feasible if the routing r_k^x does not lead to capacity violations and ensures in-time service for all already accepted requests, and in case of $\alpha_k^x = 1$ also the new request.

For our problem, the "Reward" are the lost revenues in case a request is not accepted for service. Thus, we define the reward function given state s_k and decision x_k as

$$R(s_k, x_k) = (1 - \alpha_k^x) \times R_k \tag{1}$$

3.2.4 Stochastic Information and Transition Function

After a decision is selected, stochastic information ω_{k+1} is revealed. For our problem, the stochastic information comprises a new request $\omega_{k+1} = \{c_{k+1}\}$ with the associated information on time t_{k+1}, type, locations, and revenue.

Based on state s_k, decision x_k, and stochastic information ω_{k+1}, a transition function $T(s_k, x_k, \omega_{k+1})$ leads to a new state s_{k+1}. The function sets the time to t_{k+1}. The routes r_k^x are truncated by removing all stops with arrival time smaller than t_{k+1}. Finally, the new request is c_{k+1}.

Alternatively to a new request, the stochastic information may also be an empty set ($\omega_{k+1} = \{\}$), meaning that no additional request occurs. In that case the transition function leads to the final state and the process terminates.

3.2.5 Objective Function

A solution of the problem is a decision policy $\pi \in \Pi$ with a decision rule $X^\pi : S \mapsto X_k$ that describes which decision should be chosen when the system is in a specific state, i.e. $X^\pi(s_k) = x_k$. The objective is then to find a policy that minimizes the expected total lost revenue when starting in initial state s_0:

$$\min_{\pi \in \Pi} \mathbb{E} \sum_{k=0}^{k_{max}} R(s_k, X^\pi(s_k)|s_0) \tag{2}$$

4 Approach

In the following, we present our solution approach. We first give a motivation and conceptual overview and then present the algorithmic details.

4.1 Motivation

Finding solutions for this stochastic and dynamic problem is very challenging due to the infamous curses of dimensionality (Powell 2011). Additionally, companies usually employ a larger fleet of vehicles to serve a vast number of requests per day. Furthermore, customers expect immediate responses to their requests. Thus, we propose an intuitive policy applicable to large-scale instances, letting a percentage of the fleet serve passengers with priority. The idea is related to the work by Ghiani et al. (2022) as well as to the scheduling literature where important "jobs" are given priority (see, e.g, Chen et al. 2018).

For our dynamic routing problem, we let a certain percentage of the fleet serve lucrative passenger transportation requests with priority. Transportation of good-requests are only assigned to the *priority* vehicles in case the resulting detour is very small. The remaining vehicles are free to serve both types of request. Prioritizing mobility demand for some vehicles has two purposes. First, it ensures available vehicles in case a mobility request arises. Second, it shifts transportation demand to the remaining fleet. Because the time to serve transportation requests is longer, this fosters potential consolidation opportunities with future transportation requests. Given a priority percentage, our routing heuristic iterates through all vehicles and checks whether the new request can be feasibly inserted in the existing route. In case of a transportation request and a priority vehicle, the heuristic also checks if the detour required to serve the request is below the threshold. Else, the vehicle is also indicated as infeasible. The heuristic assigns the request to the feasible vehicle with smallest detour. In case there is no feasible vehicle available, service can not be offered.

Setting a priority percentage is not trivial. If the percentage is too low, there may not be sufficient resources available to serve new mobility requests and the high revenue is lost. If the percentage is too high, priority vehicles may idle, transportation opportunities may be missed and lost revenue accumulates. Furthermore, the routes of non-priority vehicles may congest faster. As demand for passenger transportation and transportation of goods may vary over the course of the day, different percentages may be suitable for different states of the problem. The right percentage depends on a variety of factors, for example, the current workload, the ratio between mobility and transportation demand, and the expected future demand. Due to these interdependencies, analytical derivations are challenging and a derivative-free search of the solution space is required. However, the evaluation of a given percentage solution requires multiple simulation runs and is therefore quite time-consuming. This further complicates the search of a good percentage solution. To allow derivate-free search with only a limited number of evaluations, we turn to Bayesian Optimization (BO, Frazier 2018). The idea of BO is to search a vector space for a high-quality solution, especially in cases (as for our problem) where the objective function does not have a "clean" functional form and where the evaluation of a vector's objective value is time consuming. To this end, BO carefully balances exploitation of already found, good solutions and exploration of unknown areas of the vector space. While BO is still relatively unexplored in the vehicle routing literature, its advantage of finding good parametrizations within a few iterations is well-suited for this research domain where evaluating policies via simulation is quite time-consuming. For example, Dandl et al. (2021) use BO to parametrize a mobility-on-demand policy to balance revenue and social welfare.

To apply BO to our problem, we focus on one important state parameter, the point of time as it has been shown to be a good surrogate for the general state of the system (Ulmer et al. 2022). Thus, we represent the priority policy by a function based on time $p(t)$. The value $p(t) \in [0, 1]$ indicates the share of vehicles that is serving passengers with priority given a new request in time t. To apply BO, we only consider functions from a predefined space of functions, $p(t) \in \mathcal{F}_t$. We test different function spaces (Fourier series, Polynomials) of different degrees. Each function from a function space can be represented by a parametrization $a_0, \dots, a_n$ with n depending on the corresponding space and degree. Given a function space, we search the best parametrization via BO.

In the remainder of this section, we present the procedure in detail. First, we briefly describe our runtime-efficient assignment and routing heuristic given a parameter $p(t)$. We then describe the BO-procedure to determine function p.

4.2 Assignment and Routing Heuristic

For assignment and routing, we extend a runtime-efficient insertion procedure which has proven effective compared to more elaborate optimization methods in similar settings (Ulmer 2020). In its original version, given a state S_k with new request c_k, the procedure iterates through the routes $r_{k,i}$ of the vehicles to search for the most efficient, feasible insertion of both pickup and delivery location. To integrate the reservation percentage $p(t_k)$, we extend the procedure by considering both fleets differently. We apply the procedure separately for the first $p(t_k)$-percentage of vehicles V_r (priority vehicles) and for the remaining $1 - p(t_k)$, non-priority vehicles V_u (A potentially fractionally priority vehicle is treated as a normal priority vehicle). For both, the most fitting, feasible vehicle $v_r \in V_r$ ($v_u \in V_u$) with the respective additional tour duration d_r (d_u) are determined. If the new request c_k is a passenger, the request is assigned to the vehicle of both fleets with minimal extension. If the new request c_k is for transporting a good, the assignment becomes more complex. In case, the priority fleet can serve this request *very* efficiently (and $d_r < d_u$), we allow the service by the priority fleet. We model this with a maximum detour threshold $d_{\max}$ for the priority fleet. In every other case, the request is assigned to vehicle v_u if service by the non-priority fleet is feasible. Else, the request is rejected.

The algorithmic procedure for a new request in time t_k is described by Algorithm 1. First, we split the fleet V into priority V_r and non-priority part V_u given priority parameter $p(t_k)$ (line 2). We

Algorithm 1 Assignment and routing for a new requests

procedure NEWREQUEST(c_k)
    $(V_r, V_u) = \text{SPLITFLEET}(V, p(t_k))$
    $(v_r, d_r) = \text{BESTFEASIBLEINSERTION}(V_r, c_k)$
    $(v_u, d_u) = \text{BESTFEASIBLEINSERTION}(V_u, c_k)$
    $i \leftarrow null$
    if $v_u \neq$ *null* **or** $v_r \neq$ *null* **then** ▷ Service is feasible
        if $d_r < d_u$ **then** ▷ Service by priority fleet is more efficient
            if TYPE(c_k) = passenger **or** $d_r <= d_{\max}$ **then**
                $i \leftarrow v_r$ ▷ Assignment if passenger or very efficient service by v_r
            else
                $i \leftarrow v_u$ ▷ Assignment if good and no efficient service by v_r
            end if
        else
            $i \leftarrow v_u$ ▷ Assignment to no-priority vehicle v_u
        end if
        if $i \neq null$ **then** ▷ Feasible assignment found
            Accept request c_k
            Update route $r_{k,i}$
        else
            Reject request c_k
        end if
    else
        Reject request c_k
    end if
end procedure

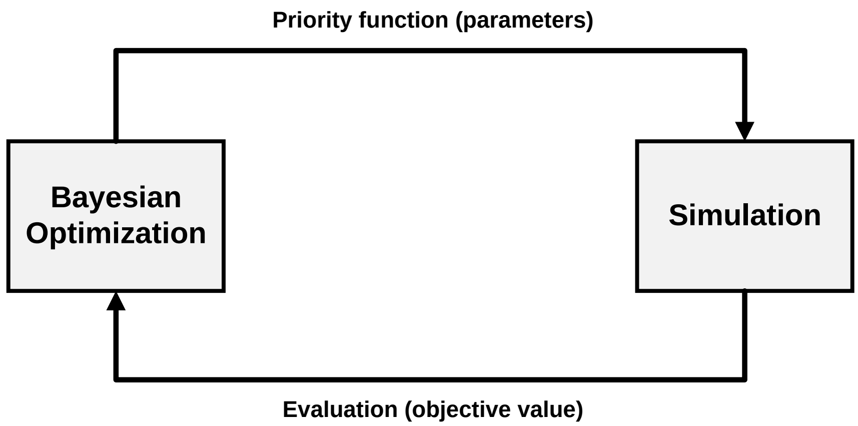

Figure 1: Interaction between Bayesian Optimization and Simulation

then determine the priority (v_r) and non-priority vehicle (v_u) with the best feasible insertion of all priority (V_r) and non-priority vehicles (V_u), i.e. the vehicle that needs the shortest detour for c_k and can still meet the deadlines of all requests assigned to them, and their corresponding needed detour d_r and d_u, respectively (lines 3-4). If a feasible insertion is found, we then choose the vehicle with the shortest detour and assign its index to variable i (lines 6-15). If a priority vehicle has the shortest detour, we need to differentiate between requests with transportation of passengers and with transportation of goods (lines 7-12). Assigning the request to a priority vehicle is only possible if c_k is either a passenger transportation request or the detour d_r does not exceed the maximum allowed detour for requests with transportation of goods ($d_{\max}$). Otherwise the best non-priority vehicle (v_u) is chosen. After choosing a suitable vehicle, request c_k is accepted and route $r_{k,i}$ of vehicle i is updated (lines 16-21). If no feasible insertion is found or the needed detour for the transportation of a good is too long, request c_k is rejected and the objective function value is updated (lines 22-24).

4.3 Bayesian Optimization

To determine the priority function $p(t)$, we apply Bayesian Optimization. The general procedure is presented in Figure 1. We iteratively apply BO and simulation of the corresponding policies. In every iteration i, BO provides us with a parametrization $a_0^i, \ldots, a_n^i$ of the priority function p^i. For evaluation, the corresponding policy π^{p^i} is simulated a large number of times, in our case, 200 times to achieve a reliable evaluation of the parametrization. This evaluation, i.e., the average objective value $V(\pi^{p^i})$ is then fed to the BO again which then determines the next set of parameters $a_0^{i+1}, \ldots, a_n^{i+1}$. We repeat this procedure 200 times ($i = 1, \ldots, 200$) to ensure convergence.

As function spaces, we apply two different types of functions, polynomials and trigonometrical functions, both often used to approximate more-complex functions via Taylor- or Fourier-series, respectively. First, we test to represent the parameter function via polynomials:

$$p(t) = \sum_{n=0}^{N} a_n \times x^n. \tag{3}$$

Second, we model the priority function via Fourier-series:

$$p(t) = a_0 + \sum_{n=1}^{N} \left(a_n \cos\left(\frac{2\pi}{t_{\max}} nt\right) + b_n \sin\left(\frac{2\pi}{t_{\max}} nt\right)\right). \tag{4}$$

For both function types, we perform a min-max-normalization to ensure the priority values are always between 0 and 1. Setting the value N is challenging, as it balances the potential of obtaining better solutions with increasing function space size by the challenge of finding them. Thus, we apply different parametrization with $N = 2, 3, 4$ leading to 3-, 4-, and 5-dimensional vectors for the polynomials and to 5-, 7-, and 9- dimensional vectors for the Fourier-series. We note that each higher degree parametrization can represent lower degrees, thus, in theory, the performance of functions with higher N should be superior. However, as our results show for both function-types, if N is too high, the solution quality decreases (again) since the solution space becomes too large. For each function-type and each value of N, we run 200 iterations of BO and evaluate each solution by 200 simulation runs of the routing heuristic. For each instance setting, we select the best found parametrization for each function-type to be applied in our computational study. We denote the corresponding policies **TD(F)** and **TD(P)** with "**TD**" indicating time-dependent priorities and "**F**" ("**P**") indicating Fourier-functions (polynomials).

For BO, we rely on the tuning provided by the *BayesianOptimization* module of GPyOpt, a open-source Python library developed by the University of Sheffield[1]. The module takes an evaluation function (in our case the simulation), a list of parameter domains and the number of iterations as input, along with some optional parameters for customization. The output of the module is the best found parametrization and the corresponding objective value. Information such as the type of the priority function (F/P) and the request data are directly passed to and handled by the simulation. The BO and the simulation were implemented in Python 3.9 using GPyOpt 1.2.5 and Java 11, respectively.

[1]https://sheffieldml.github.io/GPyOpt/

5 Computational Study

In this section, we present our computational study. We first define instances and benchmark policies. We compare the objective value of the policies and then analyze the impact of our policies on decision making. Finally, we analyze the structure of the policies and the functionality of Bayesian Optimization.

5.1 Instances

In the following, we describe our instances.

Fleet and Service Area.

We assume a 15km times 15km service area which is about the size of a medium-size city like Braunschweig, Germany. We assume Euclidean distances between locations in the service area. A fleet of 35 vehicles is assumed to operate for 10 hours, starting and ending their shift in a central depot. Vehicles travel with a constant speed of 30km per hour. Service times for pickup and drop-off are 2 minutes. Following ride-hailing provider MOIA, vehicles have a capacity of five passengers and/or goods to transport (Due to the temporal restrictions, this capacity is never reached in our experiments).

Demand.

We select demand volumes in a way that the vast majority of requests ($> 80\%$) can be served, as assumed to be realistic for this type of applications.
We set the expected number of requests to 1000 per day, equally distributed over the 10 hours. Request either comprise transportation of one person or one good. Request locations are uniformly distributed in the service area. The deadlines are set as the sum of direct travel time from pickup to drop-off plus 15 minutes for a passenger transport and plus 60 minutes for goods.
We assume that passenger transportation is significant more lucrative as transportation of goods. We model revenue dependent on travel distance. For passenger transportation, we assume 1.5 revenue units per kilometer. For transportation, we assume 0.2 revenue units per kilometer.
We generate five different ratios between passenger and good requests that allow analyzing both the value of prioritizing vehicles for passenger transportation and the performance of Bayesian Optimization. Therefore, the distributions become increasingly more complex:

1. *Constant*: In this setting, every hour 20% of requests are passenger requests.
2. *Increase*: In this setting, initially all requests are good-requests and every hour, the percentage of passenger requests increases by 10% until eventually all requests are passenger

requests.

3. *Decrease*: Similar as *Increase*, however, the day starts with 100% of passenger requests and then decreases by 10% per hour.
4. *One Peak*: One distinct peak of passenger transportation in the middle of the day. The percentages of passenger transportation over the ten hours are $0\%, 10\%, 20\%, 30\%, 40\%, 50\%, 40\%, 30\%, 20\%, 10\%$.
5. *Two Peaks*: In this setting, two (less distinct) peaks of passenger transportation are modeled. More specific, the percentages over the ten hours are $10\%, 30\%, 30\%, 40\%, 50\%, 30\%, 20\%, 30\%, 50\%, 20\%$.

5.2 Benchmark Heuristics

We compare our method to five benchmark policies divided into two sets. The first set is problem-oriented to gain insights in the benefits of combining passenger transportation and (time-dependent) priority of the fleet:

- **Split**: This policy splits the fleet with one part serving only passengers and one part only transporting goods. The best percentage per instance setting is determined by means of enumeration.

- **Myopic**: This policy assigns any feasible request to the vehicle that can serve it most time-efficiently.

- **Cost-Benefit**: This policy follows the idea of Ulmer et al. (2018a) and aims on only accepts transportation of goods, if the detour for service is below a specific time-threshold. The best threshold per instance setting is determined by means of enumeration. We denote the policy **CB**.

The second set is method-oriented to analyze the value of time-dependent priority percentages and Bayesian Optimization.

- **Fix**: This policy uses a fixed priority percentage throughout the day. The best percentage per instance setting is determined by means of enumeration in steps of 5%.

- **Continuous Approximation**: This policy is adapted from Ghiani et al. (2022) and follows the idea of finding the “right” priority percentage given a specific passenger-goods-ratio in a time period (e.g., one hour). This is done by assuming a constant ratio ($0\%, 25\%, 50\%, 75\%, 100\%$) for the entire day and finding the best **Fix**-policy for different ratios. Then,

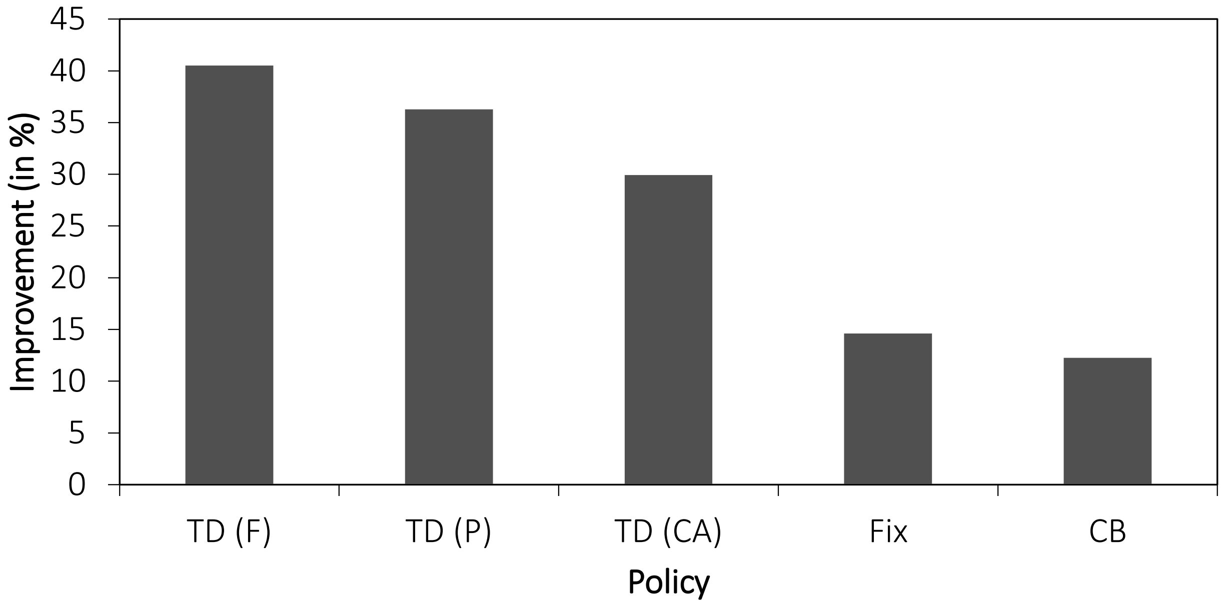

Figure 2: Average improvement compared to the **Split**-policy (for the purpose of presentation without **Myopic** (-105.0%)).

for the real instances, the expected ratio per two-hour slot is determined and the corresponding **Fix**-policy is applied. Notably, this policy neglects interdependencies between different hours of the day. We denote this policy **TD(CA)**.

5.3 Solution Quality

Figure 2 shows the average improvement of the policies over the **Split**-policy. the x-axis depicts the policies, the y-axis shows the improvement in percent. For the purpose of presentation, policy **Myopic** was omitted from the presentation as the "improvement" was -105.0%, i.e. on average, the lost revenue by **Myopic** was twice as high as for **Split**-policy. This is an important insight that a combined service of passenger and good-transportation only works in case of careful assignment strategies. We will analyze the reason for **Myopic**'s poor performance later in this section in more detail.

All other policies outperform **Split** with values between 10% and 40%. So, there is general value in combining both types of transportation. The time-dependent priority policies achieve substantially higher average improvements compared to polices **Fix** with a fixed priority percentage. This policy still outperforms the cost-benefit-policy **CB** though. From the time dependent policy, policies **TD(F)** and **TD(P)** optimized by Bayesian Optimization outperform the hand-crafted policy **TD(CA)**. The result indicate that prioritizing capacity for passenger transportation is very valuable, especially in case the priority is time-dependent. Further, since **TD(F)** and **TD(P)** outperform

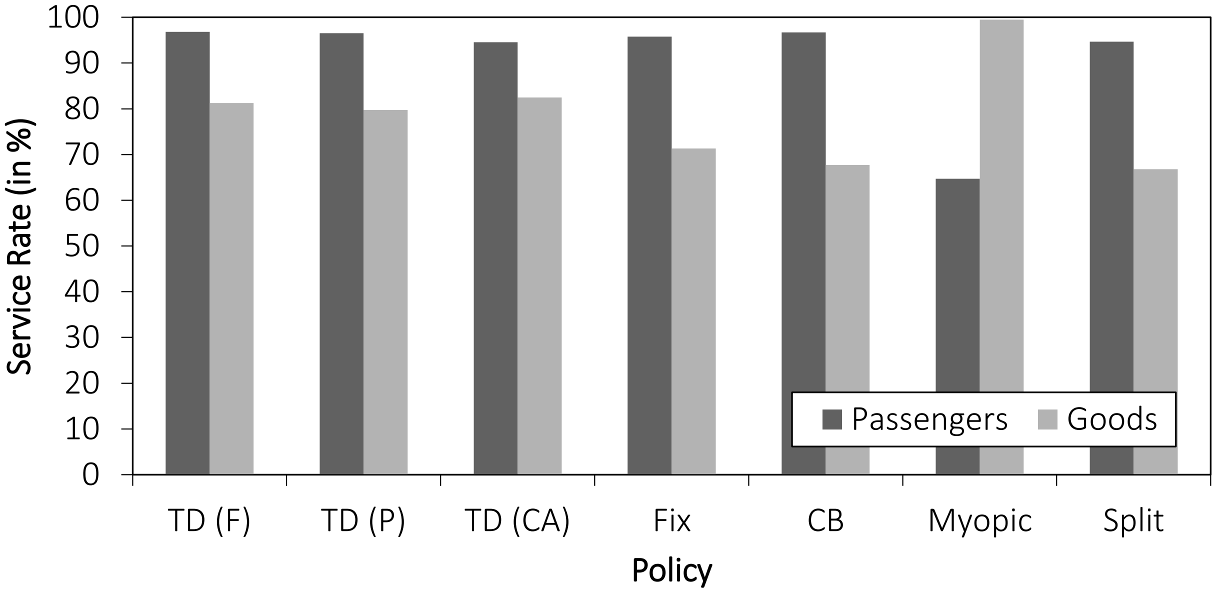

Figure 3: Average service rates for passengers and goods.

TD(CA), there is significant value in considering interdependencies in the priority percentages over the time of the day.

5.4 Service Rates

Figure 3 shows the average service rates for passengers and goods over the different policies. the x-axis depicts the policies, the y-axis shows the service rates in percent. The dark grey bars indicate service rates for passengers, the light bars service rates for transportation of goods. We observe that all policies serve the majority of requests. Policy **Myopic** is the only policy that transports more goods than passenger, likely, because passenger requests have to be declined regularly since all vehicles are busy transporting goods. Further, the policies that have been tuned have a higher service rate for passenger transportation (about 90% to 95%) than for transportation of goods (about 65% to 85%). Further, the service rate for goods differ among the tunable policies. Policies **Split** and **CB** show substantially poorer rates while especially the policies with time-dependent priorities achieve service rates above 80%.

The result indicate that policies **Split** and **CB** sacrifice transportation of good-requests to serve more passengers. This may result in additional revenue but leaves many customers unhappy. The time-dependent priority policies are able to achieve equal passenger numbers while transporting substantially more goods. Hence, there is no "cherry-picking" by the proposed TD-policies, i.e., rejecting transportation of goods-request to serve more passengers. Indeed, it is rather that the main benefit of the policies is to serve equally numbers of passengers while accommodating many more good-transports at the same time. Thus, besides increased revenue, the number of unhappy customers is reduced as well. The superior performance has three main reasons. First, the time-dependency of the priority percentage allows shifting resources between serving passengers and goods and therefore transporting goods at times demand for passenger transportation is small. Second, in contrast to a hard split between the fleets, the policies allow service of passengers by all vehicles and in some cases, even transportation of goods by priority vehicles if it can be done very efficiently. The third, more subtle reason for the increase in transportation of goods is consolidation. By prioritizing a percentage of vehicles for passengers, the transportation of goods is mainly done by a smaller set of vehicles. This increases consolidation. Indeed, about 25% of the jobs are bundled with another job for the the priority-policies, but only 12% for **Myopic** and about 9% for **CB**. Not surprisingly, the bundling percentage for **Split** is even higher with 30% as a few vehicles serve all the transportation of good-requests.

5.5 Method

In the following, we analyze the performance and structure of the tuned policies **TD(F)** and **TD(P)**. First, we show the learning process via Bayesian Optimization. Then, we analyze why (and when) approximation Fourier-functions is superior to polynomials.

Learning.

Bayesian Optimization aims on finding a valuable parametrization-vector by carefully balancing exploration and exploitation in the individual entries of the vector as we show in the following. One exemplary learning development for **TD(F)** with degree 3 and distribution *One Peak* is depicted in Figure 4. The x-axis shows the iterations, the y-axis the values. The dotted line indicate the lost revenue value of the current iteration. The solid lines shows the best-found value in the current and previous iterations.
When analyzing the best found solution, we observe a large improvement in the beginning and then stepwise improvements until convergence - this behavior is quite common for learning algorithms. Noticeably, the main convergence is reached within 100 iterations, a common observation in BO (Frazier 2018). The occasional spikes in the individual iteration's values are the result of the algorithm exploring new areas of the vector space.

Parametrization.

We apply BO for functions with different numbers of parameters, i.e., for vectors of different dimensions. With more parameters, the function space increases and therefore, better values might be found. However, at the same time, the vector-space increases and finding good values becomes

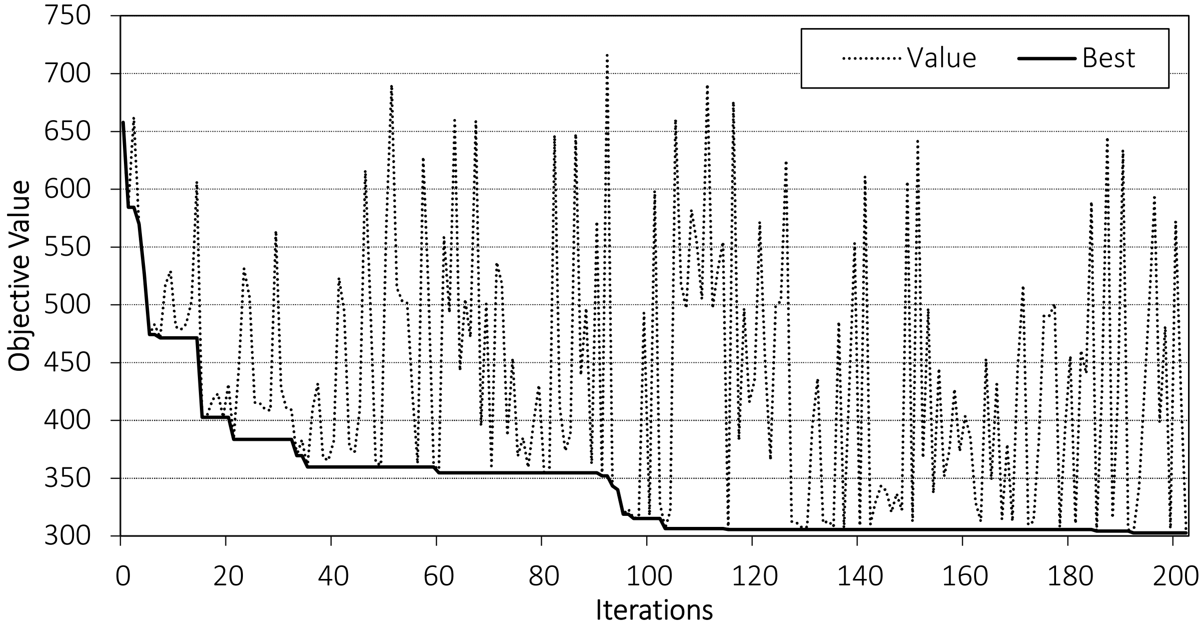

Figure 4: Individual value and best found value for **TD(F)** (degree 3) and distribution *One Peak* over the Bayesian Optimization iterations.

more challenging. We illustrate this tradeoff in Figure 5. The x-axis shows the function's degree and the y-axis shows the average improvement of the policies over **Split**. The dark grey bars represent the polynomial functions of degree 2, 3, and 4 with 3-, 4-, and 5-dimensional vectors respectively. The light grey bars indicate the Fourier-functions of degree 2, 3, and 4 with 5-, 7-, and 9-dimensional vectors.

We observe that every parametrization improves upon the **Split**-policy. For **TD(P)**, the solution quality decreases with increasing degree. Even though a higher degree includes all lower degree-solutions, the additional degree does not add to an improvement, but seems to obstruct the learning instead. Similar can be observed for **TD(F)**. However, the best parametrization can be found with a degree of 3.

Function Selection.

Our average results indicate that using Fourier-functions is advantageous compared to polynomials. In general, Fourier-functions have the advantage that their values are restricted and that they can capture local detail. In the following, we show that the latter becomes particularly important in case the demand distributions become less "clean". To this end, we analyse how the performance between **TD(P)** and **TD(F)** varies with respect to the distribution. The results are shown in Figure 6. The x-axis shows the increasingly complex demand distributions. The y-axis shows

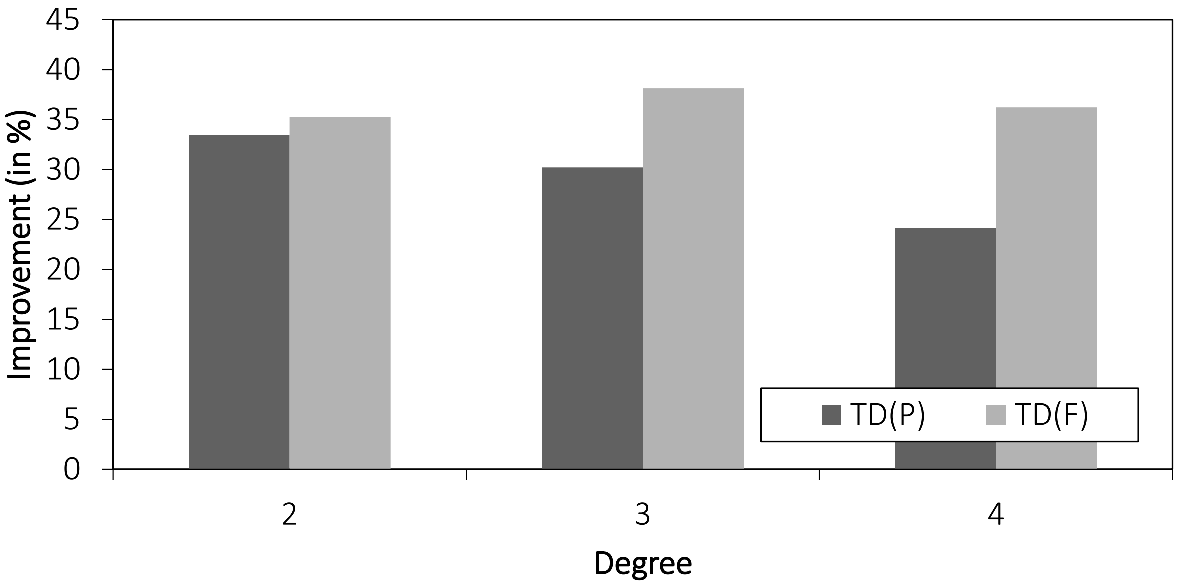

Figure 5: Average improvement for different degrees of **TD(F)** and **TD(P)**.

the average improvement of **TD(F)** over **TD(P)**. We observe that **TD(F)** outperforms **TD(P)** for all distributions except *Decrease*. The results for *Increase* and *Decrease* are rather similar for both strategies. Both distributions play to the favor of polynomial functions with clean and monotone developments in the demand ratio.

The improvements for *Constant*, *One Peak*, and *Two Peaks* are significant though. The results for the latter two distributions can be expected since they are comparably complex and therefore require a more detailed priority function. We will show this for *One Peak* later in this section. The result for *Constant* can be explained by the unrestricted nature of polynomial functions with positive degree. However, for this instance setting, even the constant priority values of **Fix** achieves similarly poor performances as **TD(P)**. This indicates that a constant priority rate may not be ideal even in case of constant demand ratio. This is linked to the dynamics of the problem, as we will illustrate by analyzing the priority values in detail in the following.

Priority Values.

For analyzing the priority values, we select distribution *One Peak* as its demand ratio-development with a clean peak in the middle of the day is more complex than linear developments but still allows an interpretation. For analysis, we plot the priority values of the best parametrizations of **TD(F)** and **TD(P)** over the service horizon in Figure 7. The x-axis shows the time in the horizon in minutes. The y-axis shows the priority percentages for **TD(F)** (solid line) and **TD(P)** (dotted line). We observe that both policies follow the general pattern of one major peak around time 300. How-

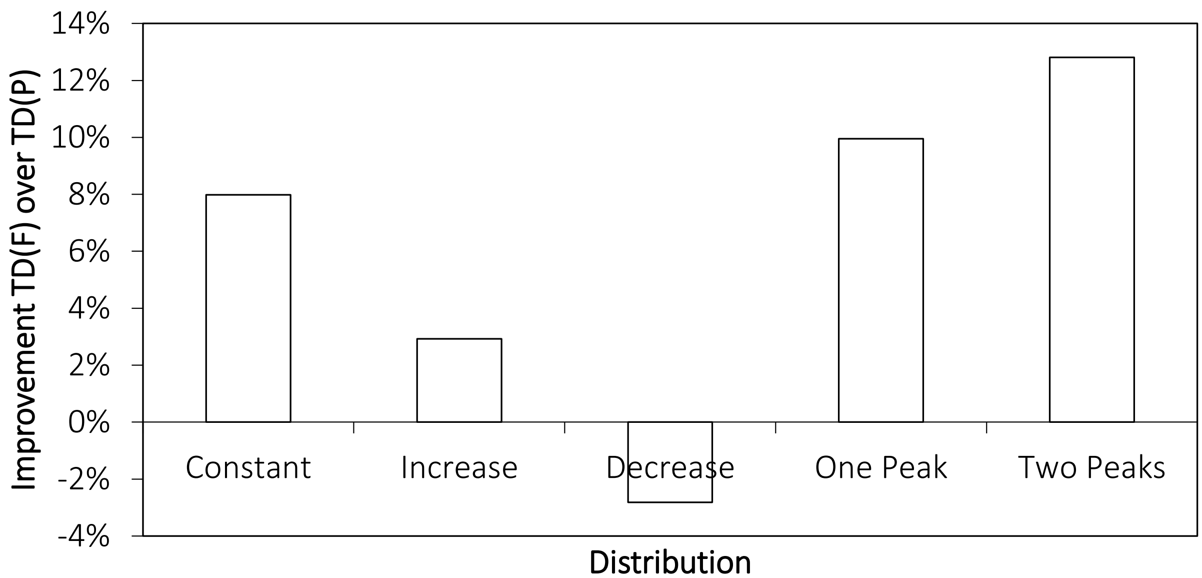

Figure 6: Average improvement of policy **TD(F)** over **TD(P)** for the five different distributions.

ever, there are differences in the details. The polynomial function essentially captures the development of the demand-ratio without any deviation. For **TD(F)**, we observe three main differences. First, the priority ratio is initially near zero and stays below **TD(P)** in the first hours. Second, the priority peak occurs slightly earlier than the passenger-demand peak. Third, the priority ratio drops faster after the peak. The explanations are as follows. First, initially, all vehicles idle, thus prioritization may be counterproductive as it prohibits efficient service. However, when resource-demanding passenger requests increase, prioritization becomes important. Second, the slightly earlier priority peak for **TD(F)** accounts for the fact that a service binds resources for a longer time (Ulmer and Savelsbergh 2020). Thus, prioritizing vehicles earlier than passenger-demand occurs ensures that the vehicle are available at time of the demand peak. The same logic explains the third difference. As priority impacts the fleet's future setup, it becomes less important when the demand-ratio decreases again. Especially, in the last hour, reserving capacity for the future is not beneficial.

6 Conclusion and Future Work

When combining mobility and transportation demand, using some vehicles with priority for mobility demand can increase revenue and - in many cases - the number of served passengers and transported goods. Varying the percentage of priority vehicles during the day can be very beneficial, especially, when the ratio between mobility and transportation demand is volatile.

There are several avenues for future work, for both methodology and problem. Until now, we used

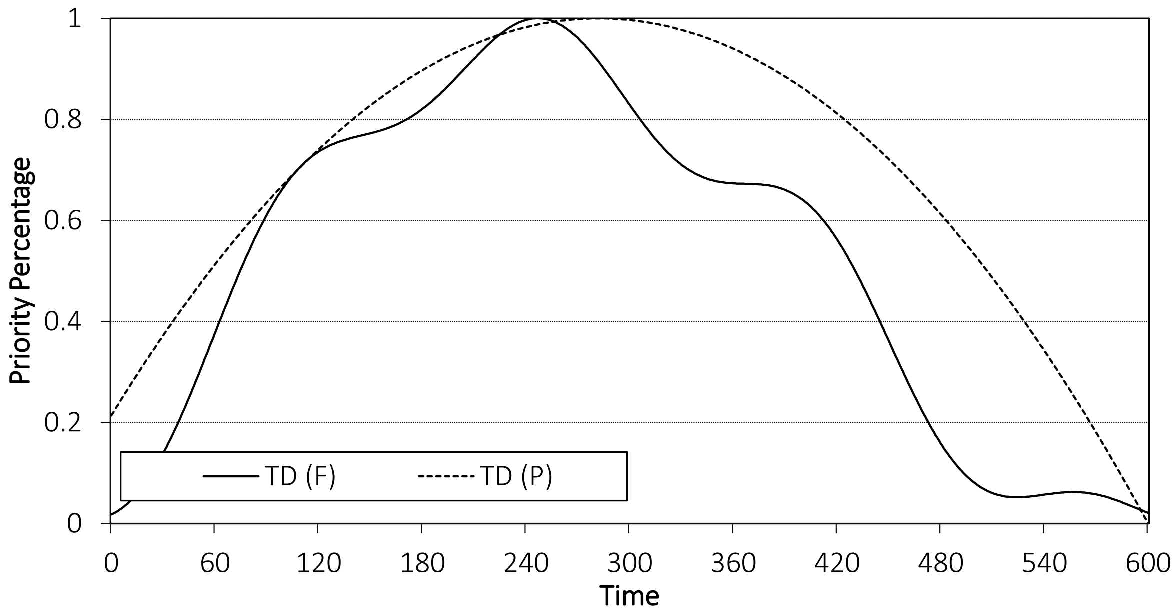

Figure 7: Average priority percentages over time for policies **TD(F)**, **TD(P)** and distribution *One Peak*.

BO to freely search for a policy with time-dependent priority. In a next steps, it might be valuable to combine BO with analytical considerations, for example, creating a promising initial solution analytically and spanning the search space around it. Another step might be to make the percentages not only dependent on time but also on other features, for example, the current workload or the vehicle distribution in the city. For the problem, it might be interesting to analyze additional services, e.g., meal delivery or transportation of elderly. In that case, a single prioritization becomes insufficient and additional measures might be considered.

References

Al-Kanj, L., Nascimento, J., Powell, W.B., 2020. Approximate dynamic programming for planning a ride-hailing system using autonomous fleets of electric vehicles. European Journal of Operational Research 284, 1088–1106.

Beirigo, B.A., Schulte, F., Negenborn, R.R., 2018. Integrating people and freight transportation using shared autonomous vehicles with compartments. IFAC-PapersOnLine 51, 392–397.

Bent, R.W., Van Hentenryck, P., 2004. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. Operations Research 52, 977–987.

Branke, J., Middendorf, M., Noeth, G., Dessouky, M., 2005. Waiting strategies for dynamic vehicle routing. Transportation Science 39, 298–312.

Brinkmann, J., Ulmer, M.W., Mattfeld, D.C., 2019. Dynamic lookahead policies for stochastic-dynamic inventory routing in bike sharing systems. Computers & Operations Research 106, 260–279.

Chen, C., Zhang, D., Ma, X., Guo, B., Wang, L., Wang, Y., Sha, E., 2017. crowddeliver: Planning city-wide package delivery paths leveraging the crowd of taxis. IEEE Transactions on Intelligent Transportation Systems 18, 1478–1496.

Chen, X., Ulmer, M.W., Thomas, B.W., 2022. Deep Q-learning for same-day delivery with vehicles and drones. European Journal of Operational Research 298, 939–952.

Chen, Y., Guo, D., Xu, M., Tang, G., Zhou, T., Ren, B., 2020. Pptaxi: Non-stop package delivery via multi-hop ridesharing. IEEE Transactions on Mobile Computing 19, 2684–2698.

Chen, Z., Demeulemeester, E., Bai, S., Guo, Y., 2018. Efficient priority rules for the stochastic resource-constrained project scheduling problem. European Journal of Operational Research 270, 957–967.

Dandl, F., Engelhardt, R., Hyland, M., Tilg, G., Bogenberger, K., Mahmassani, H.S., 2021. Regulating mobility-on-demand services: Tri-level model and bayesian optimization solution approach. Transportation Research Part C: Emerging Technologies 125, 103075.

Elbert, R., Rentschler, J., 2021. Freight on urban public transportation: A systematic literature review. Research in Transportation Business & Management , 100679.

Fehn, F., Engelhardt, R., Dandl, F., Bogenberger, K., Busch, F., 2022. Integrating parcel deliveries into a ride-pooling service–an agent-based simulation study. arXiv preprint arXiv:2205.04718 .

Ferrucci, F., Bock, S., Gendreau, M., 2013. A pro-active real-time control approach for dynamic vehicle routing problems dealing with the delivery of urgent goods. European Journal of Operational Research 225, 130–141.

Frazier, P., 2018. A tutorial on Bayesian optimization URL: `http://arxiv.org/pdf/1807.02811v1`.

Ghiani, G., Manni, A., Manni, E., 2022. A scalable anticipatory policy for the dynamic pickup and delivery problem. Submitted for publication .

Ghiani, G., Manni, E., Quaranta, A., Triki, C., 2009. Anticipatory algorithms for same-day courier dispatching. Transportation Research Part E: Logistics and Transportation Review 45, 96–106.

Ghilas, V., Demir, E., Van Woensel, T., 2016. A scenario-based planning for the pickup and delivery problem with time windows, scheduled lines and stochastic demands. Transportation Research Part B: Methodological 91, 34–51.

Goodson, J.C., Ohlmann, J.W., Thomas, B.W., 2013. Rollout policies for dynamic solutions to the multivehicle routing problem with stochastic demand and duration limits. Operations Research 61, 138–154.

Hildebrandt, F.D., Thomas, B., Ulmer, M.W., 2021. Where the action is: Let's make reinforcement learning for stochastic dynamic vehicle routing problems work! arXiv preprint arXiv:2103.00507 .

Hörsting, L., Cleophas, C., 2021. Scheduling shared passenger and freight transport on a fixed infrastructure. Available at SSRN 3886691 .

Kullman, N.D., Goodson, J.C., Mendoza, J.E., 2021. Electric vehicle routing with public charging stations. Transportation Science .

Li, L., Negenborn, R.R., de Schutter, B., 2014. Multi-agent cooperative transport planning of intermodal freight transport, in: 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), IEEE. pp. 2465–2471.

Manchella, K., Umrawal, A.K., Aggarwal, V., . Flexpool: A distributed model-free deep reinforcement learning algorithm for joint passengers & goods transportation.

Meinhardt, S., Schlenther, T., Martins-Turner, K., Maciejewski, M., 2022. Simulation of on-demand vehicles that serve both person and freight transport. Procedia Computer Science 201, 398–405.

Mitrović-Minić, S., Krishnamurti, R., Laporte, G., 2004. Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. Transportation Research Part B: Methodological 38, 669–685.

Mourad, A., Puchinger, J., Van Woensel, T., 2019. Combining people and freight flows using a scheduled transportation line with stochastic passenger demands, in: 7th INFORMS Transportation Science and Logistics Society Workshop.

Mourad, A., Puchinger, J., Van Woensel, T., 2021. Integrating autonomous delivery service into a passenger transportation system. International Journal of Production Research 59, 2116–2139.

Powell, W.B., 2011. Approximate Dynamic Programming. John Wiley & Sons, Inc, Hoboken, NJ, USA.

Powell, W.B., 2019. A unified framework for stochastic optimization. European Journal of Operational Research 275, 795–821.

Riley, C., Van Hentenryck, P., Yuan, E., 2020. Real-time dispatching of large-scale ride-sharing systems: Integrating optimization, machine learning, and model predictive control. arXiv preprint arXiv:2003.10942 .

Romano Alho, A., Sakai, T., Oh, S., Cheng, C., Seshadri, R., Chong, W.H., Hara, Y., Caravias, J., Cheah, L., Ben-Akiva, M., 2021. A simulation-based evaluation of a cargo-hitching service for e-commerce using mobility-on-demand vehicles. Future Transportation 1, 639–656.

Schilde, M., Doerner, K.F., Hartl, R.F., 2014. Integrating stochastic time-dependent travel speed in solution methods for the dynamic dial-a-ride problem. European Journal of Operational Research 238, 18–30.

Schlenther, T., Martins-Turner, K., Bischoff, J.F., Nagel, K., 2020. Potential of private autonomous vehicles for parcel delivery. Transportation Research Record: Journal of the Transportation Research Board 2674, 520–531.

Secomandi, N., 2001. A rollout policy for the vehicle routing problem with stochastic demands. Operations Research 49, 796–802.

Sheridan, P.K., Gluck, E., Guan, Q., Pickles, T., Balcıoğlu, B., Benhabib, B., 2013. The dynamic nearest neighbor policy for the multi-vehicle pick-up and delivery problem. Transportation Research Part A: Policy and Practice 49, 178–194.

Soeffker, N., Ulmer, M.W., Mattfeld, D.C., 2022. Stochastic dynamic vehicle routing in the light of prescriptive analytics: A review. European Journal of Operational Research 298, 801–820.

Tafreshian, A., Masoud, N., Yin, Y., 2020. Frontiers in service science: Ride matching for peer-to-peer ride sharing: A review and future directions. Service Science 12, 44–60.

Thomas, B.W., 2007. Waiting strategies for anticipating service requests from known customer locations. Transportation Science 41, 319–331.

Ulmer, M.W., 2020. Dynamic pricing and routing for same-day delivery. Transportation Science 54, 1016–1033.

Ulmer, M.W., Erera, A., Savelsbergh, M., 2022. Dynamic service area sizing in urban delivery. OR Spectrum , 1–31.

Ulmer, M.W., Goodson, J.C., Mattfeld, D.C., Hennig, M., 2019. Offline–online approximate dynamic programming for dynamic vehicle routing with stochastic requests. Transportation Science 53, 185–202.

Ulmer, M.W., Goodson, J.C., Mattfeld, D.C., Thomas, B.W., 2020. On modeling stochastic dynamic vehicle routing problems. EURO Journal on Transportation and Logistics 9, 100008.

Ulmer, M.W., Mattfeld, D.C., Köster, F., 2018a. Budgeting time for dynamic vehicle routing with stochastic customer requests. Transportation Science 52, 20–37.

Ulmer, M.W., Savelsbergh, M., 2020. Workforce scheduling in the era of crowdsourced delivery. Transportation Science 54, 1113–1133.

Ulmer, M.W., Soeffker, N., Mattfeld, D.C., 2018b. Value function approximation for dynamic multi-period vehicle routing. European Journal of Operational Research 269, 883–899.

Ulmer, M.W., Streng, S., 2019. Same-day delivery with pickup stations and autonomous vehicles. Computers & Operations Research 108, 1–19.

Ulmer, M.W., Thomas, B.W., Campbell, A.M., Woyak, N., 2021. The restaurant meal delivery problem: Dynamic pickup and delivery with deadlines and random ready times. Transportation Science 55, 75–100.

Voccia, S.A., Campbell, A.M., Thomas, B.W., 2019. The same-day delivery problem for online purchases. Transportation Science 53, 167–184.

Otto von Guericke University Magdeburg
Faculty of Economics and Management
P.O. Box 4120 | 39016 Magdeburg | Germany

Tel.: +49 (0) 3 91/67-1 85 84
Fax: +49 (0) 3 91/67-1 21 20

www.fww.ovgu.de/femm

ISSN 1615-4274