Bus Inf Syst Eng 59(3):161–171 (2017)
DOI 10.1007/s12599-017-0468-2

RESEARCH PAPER

On the Value and Challenge of Real-Time Information in Dynamic Dispatching of Service Vehicles

Marlin W. Ulmer · Leonard Heilig · Stefan Voß

Received: 9 June 2016 / Accepted: 1 February 2017 / Published online: 3 April 2017

Abstract Ubiquitous computing technologies and information systems pave the way for real-time planning and management. In the process of dynamic vehicle dispatching, the adherent challenge is to develop decision support systems using real-time information in an appropriate quality and at the right moment in order to improve their value creation. As real-time information enables replanning at any point in time, the question arises when replanning should be triggered. Frequent replanning may lead to efficient routing decisions due to vehicles' diversions from current routes while less frequent replanning may enable effective assignments due to gained information. In this paper, the authors analyze and quantify the impact of the three main triggers from the literature, exogenous customer requests, endogenous vehicle statuses, and replanning in fixed intervals, for a dynamic vehicle routing problem with stochastic service requests. To this end, the authors generalize the Markov-model of an established dynamic routing problem and embed the different replanning triggers in an existing anticipatory assignment and routing policy. They particularly analyze under which conditions each trigger is advantageous. The results indicate that fixed interval triggers are inferior and dispatchers should focus either on the exogenous customer process or the endogenous vehicle process. It is further shown that the exogenous trigger is advantageous for widely spread customers with long travel durations and few dynamic requests while the endogenous trigger performs best for many dynamic requests and when customers are accumulated in clusters.

Keywords Dynamic vehicle routing · Dynamic dispatching process · Stochastic requests · Real-time information · Replanning trigger

1 Introduction

In the last years, the expectations on logistic and transport service providers (SPs) have increased. Customers demand transparent, reliable, inexpensive, fast, and agile services (see, e.g., Wilding et al. 2012; Gligor and Holcomb 2014; Lowe et al. 2014). Amongst others, these services comprise transportation of parcels (see, e.g., Ulmer et al. 2016), passenger transportation (see, e.g., Mulley and Nelson 2009), and house visits of physicians or technicians (see, e.g., Chen et al. 2016). To enhance customer satisfaction, some SPs allow customers to place and update service requests at any point in time, e.g., via mobile phones (see, e.g., DHL 2013). On the operational level, SPs dispatch a fleet of vehicles to fulfill the requests. Therefore, new requests need to be dynamically assigned to vehicles considering the economic and ecologic implications of decisions (see, e.g., Hilpert et al. 2013). Since the competition

Accepted after two revisions by the editors of the special issue.

Electronic supplementary material The online version of this article (doi:10.1007/s12599-017-0468-2) contains supplementary material, which is available to authorized users.

M. W. Ulmer (✉)
Institut für Wirtschaftsinformatik, Carl-Friedrich-Gauß Fakultät, Technische Universität Braunschweig, 38106 Braunschweig, Germany
e-mail: m.ulmer@tu-braunschweig.de

L. Heilig · S. Voß
Institut für Wirtschaftsinformatik, HBS Hamburg Business School, University of Hamburg, 20146 Hamburg, Germany
e-mail: leonard.heilig@uni-hamburg.de

S. Voß
e-mail: stefan.voss@hamburg.de

in related markets is often high, effective routing and assignment decisions may lead to a valuable competitive advantage with respect to responsiveness and service costs.

In their routing and assignment decisions, dispatchers nowadays can draw on real-time information about new requests and the vehicle statuses and locations. Savelsbergh and Van Woensel (2016) state that "embedding and effectively using high-quality information [...] in decision support (systems)" is "critical, but nontrivial". For the dispatching of vehicles, dispatchers need to determine suitable times to replan while considering a tradeoff between efficient routing and effective assignments. Frequent replanning may allow for efficient routing since vehicles can be instantly diverted from the current destination. Less frequent replanning may lead to an accumulation of requests and more effective assignment decisions. The resulting questions are summarized by Speranza (2016): "When should a change in the data imply a rerun of a model? [...] Which changes in the data make the rerunning of a model beneficial?" In this paper, we address these questions by analyzing the impact of the three most commonly used triggers for replanning from the literature:

1. *Exogenous customer process* Replanning is triggered by a new customer requesting service. This allows a diversion of current routes leading to efficient routing. Still, this may lead to premature assignment decisions and drivers' inconvenience because they need to change their destination while en route.
2. *Endogenous vehicle process* Replanning is triggered by the vehicle just finished serving a customer. This allows exploiting the gain in information leading to effective assignment decisions. Still, it may lead to inefficient routing and to customers' inconvenience because they may need to wait for information.
3. *Fixed intervals* Replanning is conducted in predefined time steps. This can be seen as a combination of the advantages and shortcomings of the first two triggers.

To analyze the impact of the triggers, we generalize an existing Markov-model of the dynamic *vehicle routing problem with stochastic service requests* (VRPSSR) to allow decision making based on the three triggers. We further modify an existing assignment and routing approach by Ulmer et al. (2016) to the new model. We conduct extensive experimental studies for varying instance settings. Our analysis provides the following implications:

- Replanning triggered by exogenous and endogenous processes are generally superior to planning in fixed intervals.
- The exogenous trigger is particularly beneficial for instance settings with widely spread customers and few dynamic requests.
- The endogenous trigger is particularly beneficial for instance settings with clustered customers and many dynamic requests.

Our contributions are as follows. This paper presents the first structured analysis of how the three major replanning triggers impact solution quality. To this end, we present a comprehensive BPMN (Business Process Model and Notation) model of the dispatching process integrating both the exogenous customer process and the endogenous vehicle process. We further generalize both an existing Markov decision process model and a state-of-the-art solution method to capture the different decision triggers. We finally derive guidelines indicating which trigger is beneficial in which instance characteristics.

The remainder of this paper is structured as follows. In Sect. 2, we present the BPMN model, define the three decision triggers, and present a literature classification based on how the papers approach and model decision making with respect to these triggers. Section 3 formally defines the VRPSSR and gives an example of the Markov decision process model. In Sect. 4, we present the generalization of the assignment and routing approach by Ulmer et al. (2016). In an extensive computational study, the approaches are compared in detail for a variety of real-world sized instances in Sect. 5. The paper concludes with a summary of the results and directions for future research in Sect. 6.

2 Vehicle Dispatching

In this section, we present a BPMN model to define the considered dispatching processes. More specifically, we depict important activities, information flows between the individual processes of involved actors (i.e., customers, dispatchers, and service employees), and the three aforementioned decision triggers. Moreover, this section presents an overview and analysis of related literature.

2.1 Business Process Model

In this section, we present a vehicle dispatching process and explain the ways real-time information can be used for decision making. The business process is highly dependent on a central information platform and mobile technologies allowing an ubiquitous connectivity of service employees and real-time information exchange. In Heilig (2017), a mobile cloud platform, referred to as *port-IO*, has been specifically developed to support such business processes by implementing interfaces for gathering real-time data (e.g., position and status of vehicles, traffic data) and incorporating it into an advanced cloud-based decision support component used to optimize routes and synchronize the results with respective service employees. For such applications, the analysis of the value of real-time information and decision making processes is essential for effective decision making.

As depicted in Fig. 1, the process model represents the flow of activities and information among customers, dispatchers, and service employees using BPMN (for the latest specifications, the interested reader is referred to OMG 2013). Before we explain the modes of decision making in this business process, we describe the main planning and operational activities.

At any time, customers are able to place and update service requests. After storing new or updated requests in the information system of the SP, a process for dispatching a fleet of vehicles to fulfill these requests is triggered. To consider the actual position and status of service employees (i.e., vehicles), available contextual data is synchronized between vehicles and the information system of the SP. The real-time exchange of information is realized through a mobile app executed on the employees' mobile device. Moreover, other sources of information, such as traffic control systems for receiving current traffic data, may be queried before starting the planning and optimization of vehicle routes. After receiving relevant data and running the optimization, the planned schedules are passed to the respective service employees via the mobile app. In this regard, two different situations may occur. If a new tour has been assigned to an available service employee, a new service process is initiated. Otherwise, in case a current tour receives updates, the respective process is interrupted to consider the changes received, allowing an immediate diversion from the current plan. Note that the updates can only be received after fulfilling the last service request in case the employee has already arrived at the respective customer's location. Therefore, the process is interrupted while driving to the next customer or, if the update is already available, before starting a new trip to the next customer, as depicted in the BPMN model. After the service employee has checked new instructions, the mobile app navigates the service employee to the next customer according to the new list of requests. Meanwhile, the dispatcher has informed the customer about the planned completion date and time. These activities are repeated until all service requests assigned to the vehicle are fulfilled and the end of the shift is reached.

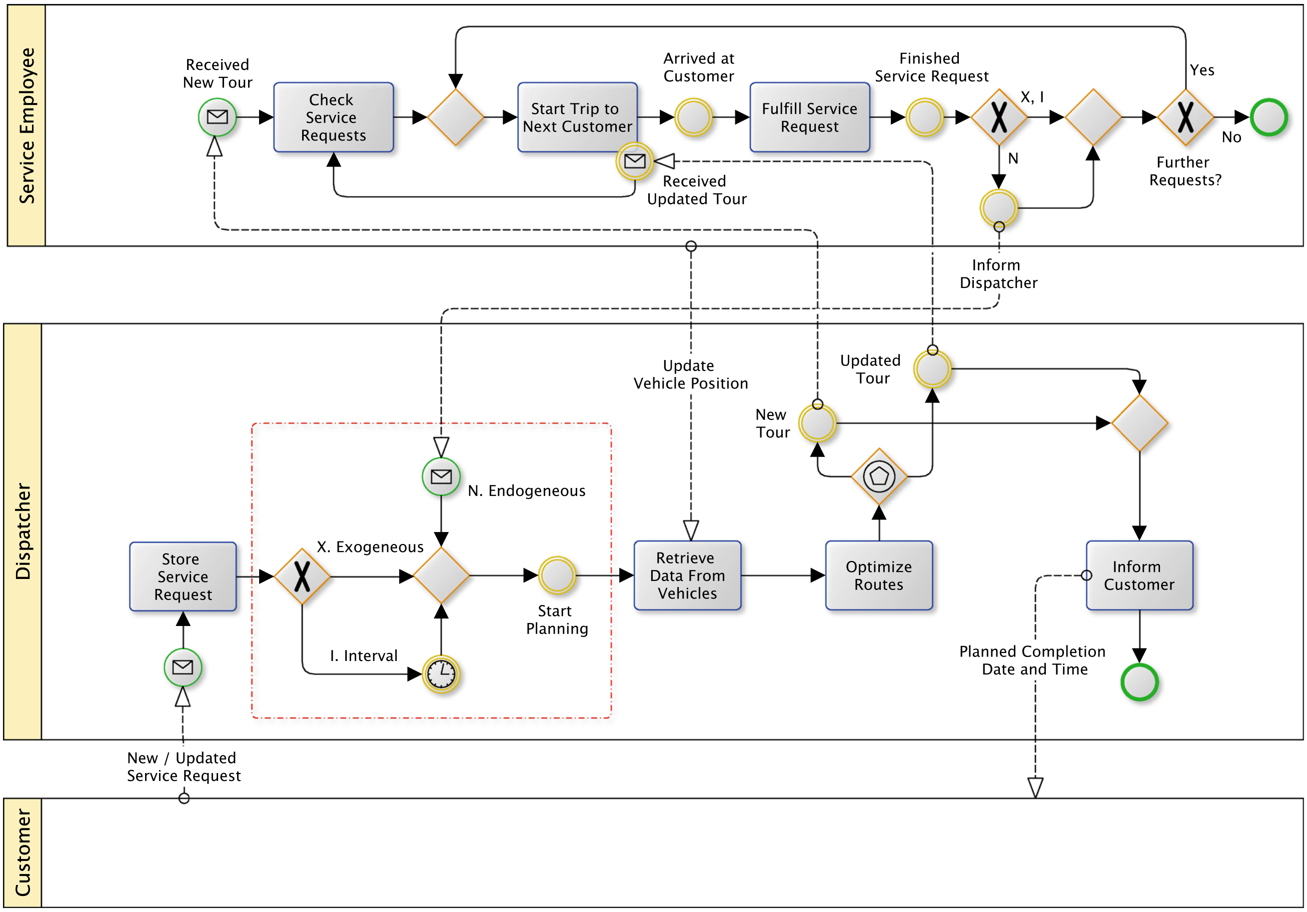

Fig. 1 BPMN model of the dispatching process

In the presented business process, three modes of decision making are possible. We refer to them as exogenous, endogenous, and interval-based decision making. In the process model, the associated process events are framed in red. As we will assess the value of each of those modes in this paper, we briefly explain the differences in the following.

- *X. Exogenous decision making* A new or updated request is immediately handled by the dispatcher. That means that the dispatcher directly incorporates the new request by replanning current activities of the fleet. Once a new plan has been determined, at least one service employee receives an update of the current plan to serve additional customers. The main aim of this mode is to allow an immediate diversion and efficient routing.
- *N. Endogenous decision making* Each time the service employee served a customer (i.e., location specified in the service request), the dispatcher is informed. In this moment, the dispatcher checks if new, unassigned requests exist and, if so, checks whether it is reasonable to assign them to the service employee. That is, instead of planning each service request immediately (mode X), the dispatcher may benefit from a gain of information and bundle service requests arrived during the time the service employee drives to the next customer for more effective assignments.
- *I. Interval-based decision making* The dispatching process is triggered repeatedly after a fixed time interval. This traditional approach allows to bundle several service requests which have arrived in the meantime. The approach neither considers the current context of the service employees (mode N) nor of requesting customers (mode X).

In this paper, we focus on the impact of the three replanning triggers and analyze in which environments they are particularly beneficial. To this end, we focus on a service employee or a single vehicle, respectively.

2.2 Related Works

In this section, we present related literature on dynamic dispatching problems with stochastic requests. The work on dynamic dispatching problems is vast. For an extensive survey, the interested reader is referred to Ritzinger et al. (2016), Oyola et al. (2016a, b). The literature is presented in Table 1. We focus on work with respect to three dimensions. First, we analyze *when* replanning is triggered. We differentiate between fixed intervals (I), the endogenous routing process (N), and the exogenous request process (X). We further analyze whether the model allows for diversion, and whether the solution approach explicitly incorporates stochastic information to achieve anticipation.

The works differ with respect to the point in time, in which decisions are triggered. As examples, the approach in Regan et al. (1996) decides when a new request occurs analyzing the impact of diversions. In Thomas (2007),

Table 1 Literature classification

	Decision trigger (X, N, I)	Diversion	Anticipation
Regan et al. (1996)	X	✓	
Gendreau et al. (1999)	X		
Ichoua et al. (2000)	X	✓	
Gutenschwager et al. (2003)	X, N		
Bent and Van Hentenryck (2004)	X		✓
Thomas and White (2004)	(N)	✓	✓
Chen and Xu (2006)	I	✓	
Hvattum et al. (2006)	I		✓
Thomas (2007)	N		✓
Hvattum et al. (2007)	I		✓
Ichoua et al. (2006)	N		
Angelelli et al. (2009)	I	✓	
Ghiani et al. (2009)	X		✓
Branchini et al. (2009)	X	✓	
Lorini et al. (2011)	X, (N)	✓	
Ghiani et al. (2012)	N		✓
Sheridan et al. (2013)	X, N	(✓)	✓
Ferrucci and Bock (2015)	I	✓	
Ulmer et al. (2016)	N		✓
Our Work	X, N, I	✓	✓

decisions are made when the vehicle is located at a customer, and in Hvattum et al. (2007), replanning is conducted in intervals of 10 min and 3 h. In Lorini et al. (2011), the process is usually triggered by new requests but, since travel times are stochastic, in some cases also by vehicles' delay. Gutenschwager et al. (2003) decide whenever a vehicle idles or when a new customer requests service.

In many papers, the possibility on diversion is connected to the decision trigger. Usually, diversion is only possible in cases where decisions are triggered by new requests or conducted in fixed intervals. In the special case of Thomas and White (2004), decision points arise when the vehicle finishes travel of a street segment. Hence, the decision is triggered by the endogenous process but diversion is allowed. Sheridan et al. (2013) consider an assignment problem for taxis. In this case, replanning is triggered when a vehicle is free or a new customer request comes in. In both cases, vehicles may be reassigned. This does not result in an explicit diversion, indicated by "(✓)" in the diversion-field. Ferrucci and Bock (2015) decide about diversions based on the resulting inconvenience of drivers and customers. Gutenschwager et al. (2003) do not allow diversion. To avoid inefficient assignments, they develop strategies for postponing assignments to exploit the gain in information over time.

Some approaches, such as Gendreau et al. (1999), only draw on the currently available information in their decision making without anticipation. Still, anticipation of future requests improves decision making in many cases. Anticipation is often achieved by means of approximate dynamic programming (ADP, Thomas and White 2004; Thomas 2007; Ghiani et al. 2009, 2012; Ulmer et al. 2016). A multiple scenario approach (MSA) is used to anticipate requests in Bent and Van Hentenryck (2004); Hvattum et al. (2006, 2007). The MSA samples a set of new requests in real-time and evaluates routing plans with respect to the samples. Ulmer et al. (2016) present an offline ADP-approach to decide about assignments. Offline approaches are highly valuable since they conduct the majority of the calculations before the actual implementation of the algorithm. This allows an immediate communication either to the drivers or the customers. This is particularly important to exploit the benefit of diversions. Hence, we adapt the approach from Ulmer et al. (2016) for our computational study.

3 Problem Definition: The VRPSSR

In this section, we define the VRPSSR and give an example of the Markov decision process (MDP) model (Puterman 2014). For the comprehensive MDP model of the VRPSSR, the interested reader is referred to Sect. A.1 in the Appendix (available online at http://link.springer.com).

3.1 Problem statement

The required notation for the VRPSSR is depicted in Table 2. The problem is a generalization of Ulmer et al. (2016). A vehicle serves customers in a service area $\mathcal{A}$ considering a continuous shift $T = [0, t_{\max}]$. The tour starts and ends in a depot $D \in \mathcal{A}$. A set of known early request customers (ERCs) is assumed to be given in $t = 0$ and must be served. In the beginning of the day, the dispatcher determines a planned tour θ_0 visiting all ERCs. A planned tour consists of the sequence of customers and the related arrival times. At every point in time during the shift, the vehicle's location l is known.

During the shift, new stochastic late request customers (LRCs) request service. Notably, the LRCs are unknown until their time of request. All requests follow a spatial distribution $\mathcal{F}$ mapping customers in the service area $\mathcal{A}$. The deterministic travel duration between two customers C_1, C_2 and/or the vehicle's current location is defined by $d(C_1, C_2)$. The overall travel duration of the current tour θ starting in the vehicle's current location l is $\bar{d}(l, \theta)$. Each (unknown) LRC C has a request time t^C. The request time t^C follows a temporal stochastic distribution $t^C \sim \mathcal{T}$. Service times are neglected meaning that a customer is served upon arrival of the vehicle.

A replanning is triggered when a new customer request is received (X), the vehicle served a customer (N), or a specific point in time is reached (see process description in Fig. 1). This point in time was previously determined by the dispatcher and models interval decision making (I). The dispatcher decides about the set of open LRCs, the update of the current routing plan, and about the next time a replanning is considered. Assignment decisions determine, on the one hand, whether to assign or reject a request, and

Table 2 Notation: Problem Statement

Description	Notation
Service area	$\mathcal{A}$
Shift	$T = [0, t_{\max}]$
Depot	$D \in \mathcal{A}$
Planned initial tour	θ_0
Current vehicle location	l
Spatial request distribution	$\mathcal{F}$
Travel duration between C_1, C_2	$d(C_1, C_2)$
Travel duration of location l and tour θ	$\bar{d}(l, \theta)$
Request time of late request customer C	t^C
Temporal request distribution	$\mathcal{T}$

on the other hand, to postpone the decision about the request. If the dispatcher decides to assign the request, it is possible to change the overall tour θ accordingly. A tour is feasible as long as the duration of the new tour does not violate the time limit meaning that the vehicle is not allowed to arrive at the depot later than $t_{\max}$. Waiting at customer locations is permitted and diversion from the current destination of the vehicle is possible. Since the resources are limited, the dispatcher aims at utilizing the limited resources to maximize the expected number of assigned (and therefore fulfilled) LRCs over the day.

3.2 Example: Markov Decision Process

In the following, we illustrate the Markov decision problem by means of an example. We also introduce the notion of a *post-decision state* and the *free time budget*, since these are utilized by the solution algorithms. Figure 2 depicts a decision state S_k, a decision x, and a post decision state S_k^x. The current point in time is $t_k = 120$. The left side of Fig. 2 shows the state S_k. The diamond depicts the current location of the vehicle, the circles the customers, and the square the depot. The light circles $\{C_1, C_3, C_4\}$ indicate assigned customers, the dark circles open requests $\{C_2, C_5, C_6\}$. The gray circle C_0 indicates the vehicle's origin and is only depicted for the purpose of presentation. Overall, six customers are currently assigned or open. The vehicle currently travels to customer C_1. The current sequence is C_1, C_3, C_4, D. The according arrival times are not depicted in Fig. 2, but presented in the following. The vehicle currently arrives at customer C_1 in 145, at customer C_3 in 180, at customer C_4 in 200, and at the depot D in 240. Hence, the current planned tour is $\theta_{t-1} = (C_1 : 145, C_3 : 180, C_4 : 200, D : 240)$. The overall time limit for this example is $t_{\max} = 360$.

For the current tour θ_{k-1}, the dispatcher has a free time budget of $b_k = 360 - 240 = 120$ min to assign additional customers. Decision x now assigns customer C_5, rejects C_6, and postpones the decision about C_2. The reward of the decision is 1 since one new customer is assigned. Customer C_6 cannot be included efficiently in the current tour and is therefore rejected for this vehicle. Customer C_5 is close to the current vehicle location, and the vehicle diverts from the current destination to serve C_5. The assignment decision for customer C_2 is postponed as the customer is not close to the tour; however, this may be the case later since the vehicle visits the neighborhood of customers C_4 and C_2 at later points of time in the shift. The dispatcher can wait for future developments before deciding about C_2. The resulting post-decision state S_{120}^x is depicted on the right side of Fig. 2. The new planned tour θ_k is only a mild update of θ_{k-1} by including C_5 but maintaining the remaining sequence. The arrival time for C_5 is 130. The arrival times for C_1, C_3 and C_4 are shifted by the insertion of C_1 by 20 minutes. The new arrival times are 165, 200, and 220, respectively. The vehicle will currently arrive at the depot at 260. The updated plan is $\theta_t = (C_5 : 130, C_1 : 165, C_3 : 200, C_4 : 220, D : 260)$. The new free budget is $b_k^x = 360 - 260 = 100$ minutes. Finally, the dispatcher sets the next potential decision point to 130. The next realized decision point occurs either at time 130 or earlier if a new customer service request occurs.

Notably, the MDP allows decision making for all three triggers. This generalization is necessary to apply all approaches to the same problem model. Still, the three approaches only consider *one* trigger each. This means that in the MDP-context, decision points of "wrong" triggers are ignored by postponing all requests and maintaining the planned tour.

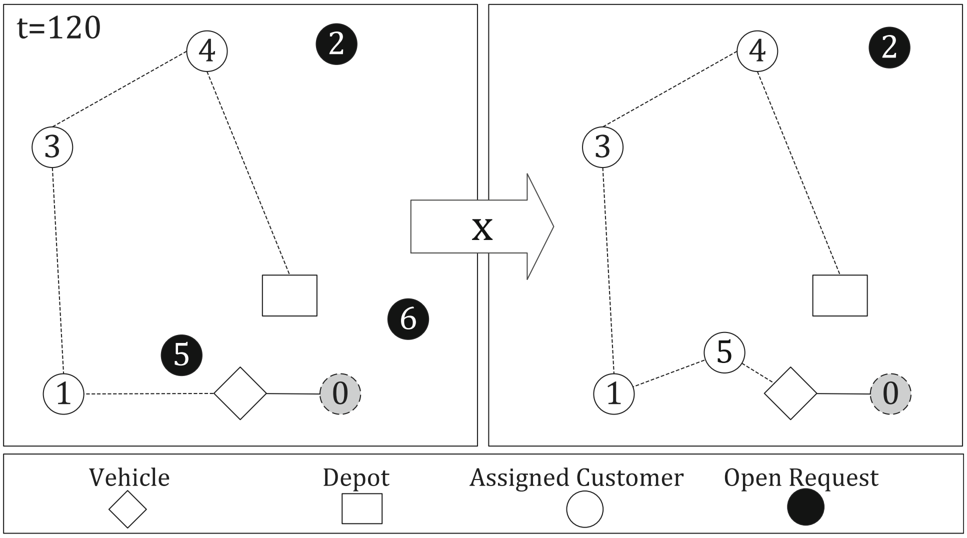

Fig. 2 Example: state, decision, and post-decision state

4 Solution Approach

In this section, we present the applied solution approaches. We focus on the assignment decisions, since Ulmer et al. (2016) show that the impact of assignments for a similar dynamic routing problem is significantly higher than the impact of the routing decisions. We define the assignment policy ATB (Anticipatory Time Budget) by Ulmer et al. (2016) in Sect. 4.1. For routing, we draw on the cheapest insertion heuristic (CI; Rosenkrantz et al. 1974). Newly assigned requests are included in θ_{k-1} via CI. To this end, CI maintains the sequence of θ_{k-1} and subsequently inserts the requests cheapest to integrate in this sequence. If all assigned customers are served and some free time is still left, the vehicle idles at its current location either until a new request is assigned or it returns to the depot. The solution approaches vary in the point in time, in which a replanning is triggered. Therefore, we modify ATB to consider different replanning triggers.

4.1 Assignments: Anticipatory Time Budget

In this section, we recall ATB by Ulmer et al. (2016). We apply the same procedure for all three triggers differing in the time of the decision. At a respective decision point, the power set of the accumulated open requests is generated. Each subset of open requests is first checked for feasibility and then evaluated by its potential of serving future customers. The customers of the best subset are assigned. All the remaining open requests are rejected. We do not consider postponements, because they lead to an exponential increase of the decision space due to the subset selection (see Powell 2011). Since decisions need to be derived in real-time, rejections limit the decision space and allow derivation of decisions within reasonable runtimes. In the following, we describe how ATB determines the subset to assign by means of approximate dynamic programming.

The objective of the VRPSSR is to find a decision policy $\pi \in \Pi$ maximizing the expected number of assignments. A policy is a function mapping a state S_k to a decision $X^{\pi^*}(S_k)$. A decision determines the subset of requests to assign and the update of the routing. In a particular decision state S_k, the optimal policy $\pi^* \in \Pi$ fulfills the Bellman Equation:

$$X^{\pi^*}(S_k) = \operatorname*{arg\,max}_{x \in X(S_k)} \left\{ R(S_k, x) + V(S_k^x) \right\}. \tag{1}$$

Here the value $V(S_k^x)$ represents the expected number of assignments starting from post-decision state S_k^x following policy π^*. Hence, if we have access to the value of each post-decision state, we would achieve an optimal policy. Still, the exact values cannot be derived due to the "Curses of Dimensionality" describing the exponential increase in model-complexity in state-space, decision-space, and transition-space (see Powell 2011). Hence, ATB approximates the values by means of simulation.

To this end, ATB aggregates post-decision states to vectors of the current point in time t_k and the related free time budget b_k^x. The lower the point in time, the more customers will request service in the future and the more can be assigned. The higher the time budget, the more customers can be assigned to the vehicle in the future. ATB uses an aggregation function $\mathfrak{A} : S_k^x \rightarrow v$ mapping a post-decision state to a two-dimensional vector $v = (t, b_t^x)$.

ATB approximates the values $\hat{V}(v)$ for each vector v by means of approximate value iteration (AVI; Powell, 2011, pp. 391ff). First, AVI assigns initial values $\hat{V}_0(v)$ to every vector v. These values induce a policy π_0 with respect to the Bellman Equation as depicted in Eq. (2) for $i = 0$,

$$X^{\pi_i}(S_k) = \operatorname*{arg\,max}_{x \in X(S_k)} \left\{ R(S_k, x) + \hat{V}_i(\mathfrak{A}(S_k^x)) \right\}. \tag{2}$$

AVI subsequently simulates realizations $\omega_1, \ldots, \omega_m \in \Omega$. Within the simulation of realization ω_i, the current policy π_{i-1} is applied according to Eq. (2). After each simulation run, the observed values are updated according to the realized values $\hat{V}_{\omega_i}$ within the simulation run as shown in Eq. (3):

$$\hat{V}_i(v) = (1 - \alpha)\hat{V}_{i-1}(v) + \hat{V}_{\omega_i}(\mathfrak{A}(S_k^x)). \tag{3}$$

Parameter α defines the stepsize of the approximation process. With the number of observations of v, $N(v) > 0$, ATB draws on $\alpha = \frac{1}{N(v)}$, i.e., the running average over all observed values for v. For approximation, we run simulations for 1 million realizations. To enforce the approximation process, we apply the dynamic lookup table proposed by Ulmer et al. (2016), dynamically partitioning the vector space.

4.2 Replanning Trigger

To compare the previously defined decision triggers, we define three policies, ATB^{X}, ATB^{N}, ATB^{I} differing in the decision points at which the assignment decision is conducted and the routing is updated, accordingly. Policy ATB^{X} decides at each point in time a new customer request occurs. Except in the rare case of two customers requesting at the same time, ATB^{X} only considers two possible post-decision states. One post-decision state results from the potential assignment of the new request. The other results from the rejection. Policy ATB^{X} exploits the potential of diversion and efficient routing since the vehicle may be on the road at time of the new request.

Policy ATB^{N} decides in every point in time, the vehicle is located at a customer. In some cases, the set of accumulated open requests may therefore be empty resulting in only one potential post-decision state. In other cases, the cardinality of the set is high and the number of post-decision states and potential decisions is vast. Policy ATB^{N} therefore exploits the information gain of accumulated requests allowing effective assignments. Still, ATB^{N} is not able to divert the vehicle.

Finally, ATB^{I} decides in fixed intervals of time and can be seen as a compromise between ATB^{X} and ATB^{N}. As ATB^{N}, ATB^{I} accumulates requests over the interval and achieves a gain in information. As ATB^{X}, ATB^{I} decides when the vehicle is on the road and therefore enables diversions. In our computational study, we analyze intervals of 5, 10, 15, 20, and 30 min.

4.3 Benchmark: Myopic Policy

We assume that the benefit of both diversion and the gain in information depends on the assignment policy. To verify this assumption, we compare the ATB policies with the myopic equivalent. In every active decision point, the myopic policy maximizes the immediate reward assuming post-decision state values of zero. In cases, where several decisions $X^{\max} = \{x_1, \ldots, x_m\}$ lead to the same immediate reward, the myopic policies select the decision $x \in X^{\max}$ maximizing the remaining free time budget.

5 Computational Experiments

In this section, we motivate and define the test instances and test measures. Then, we present the solution quality of the different policies. Finally, we analyze the policies' performance with respect to the instances' dimensions. The algorithms are implemented in Java. We run the tests on Windows Server 2008 R2, 64 bit, with an Intel-Xeon E7-4830@2.13GHz, 64 cores, and 128GB RAM. We omit further details on implementation and runtime aspects because the simulations are conducted offline. The runtimes in the online execution of the algorithms are generally less than one millisecond per decision point.

5.1 Instances

In the following, we define the instance settings. To analyze the impact of the different triggers in different settings, we define instances varying in the number of requesting customers and the travel durations between customers. To this end, we vary the overall expected number of requests as well as the percentage of dynamic requests, the *degree of dynamism* (DOD). We further analyze three different customer distributions and three differently sized service areas.

We vary the expected number of customers between $n = 20, 30, 40, 60$ for a DOD of 0.25, 0.5, 0.75. An expected number of customers $n = 40$ and a DOD of 0.75 results in an expected ERC number of $n_0 = 40 \times (1 - 0.75) = 10$ and an expected LRC number of $n_+ = 40 \times 0.75 = 30$.

We define a small service area $\mathcal{A}_{10}$ with a size of 10 km × 10 km, a medium-sized $\mathcal{A}_{15}$, and a large service area $\mathcal{A}_{20}$, with sizes of 15 km × 15 km and 20 km × 20 km, respectively. Finally, we define three spatial customer request distributions $\mathcal{F}$. We test the policies for uniformly distributed requests (U) as well as for requests in two ($2C$) and three clusters ($3C$). In the following, we define the clustered distributions for $\mathcal{A}_{20}$. For the other service areas, the parameters are multiplied by $\frac{1}{2}$ and $\frac{3}{4}$. For $2C$, the cluster centers are (5, 5) and (15, 15). For $3C$, the cluster centers are (5, 5), (5, 15), and (15, 10). Within the clusters, the request probability is normally distributed with standard deviation of $1km$.

The remaining parameters are defined as follows. The overall time limit is $t_{\text{limit}} = 360$ minutes. The temporal distribution of requests $\mathcal{T}$ follows a Poisson process. The depot D is located in the center of the service area and the vehicle travels with a constant speed of 15 km/h. The combination leads to an overall set of 108 different instance settings $\mathcal{I}$. We run 10,000 test runs per instance setting $i \in \mathcal{I}$.

5.2 Measures

To allow an analysis of the results, we define two measures, namely solution quality Q and improvement Δ. These measures reflect the solution quality of individual policies but also allow comparison between two approaches. In the following, we define the measures for the analysis of the results. Let $\mathcal{R}(\pi, i)$ be the average number of served requests for policy π and instance setting $i \in \mathcal{I}$. Further, let $n_+(i)$ denote the overall number of expected LRCs for instance setting i. We define the solution quality $Q(\pi, i)$ of a policy π for an instance setting i as the average percentage of served requests as depicted in Eq. (4),

$$Q(\pi, i) = \frac{\mathcal{R}(\pi, i)}{n_+(i)}, \tag{4}$$

$Q(\pi, I)$ is then defined as the average solution quality over all instances $i \in I \subseteq \mathcal{I}$. $Q(\pi)$ denotes the average solution quality over all instance settings $\mathcal{I}$. We further define the *improvement* $\Delta(\pi_1, \pi_2, i)$ of policy π_1 over policy π_2 given instance setting $i \in \mathcal{I}$ as the percentual improvement of solution quality $Q(\pi_1, i)$ compared to $Q(\pi_2, i)$ as depicted in Eq. (5),

$$\Delta(\pi_1, \pi_2, i) = \frac{Q(\pi_1, i) - Q(\pi_2, i)}{Q(\pi_2, i)}. \quad (5)$$

The improvement $\Delta(\pi_1, \pi_2)$ denotes the average of the individual improvements over all instance settings $\mathcal{I}$.

5.3 Results

In this section, we compare the results of the different policies. We focus on general statements with respect to the instance setting's dimensions. For the individual results, we refer to Sect. A.2 of the Appendix (available online at http://link.springer.com). The average solution quality over all instance settings for the different policies is shown in Fig. 3a. The best results are achieved by ATB^X with $Q(\text{ATB}^\text{X}) = 58.3\%$ assignments, followed by ATB^N with $Q(\text{ATB}^\text{N}) = 58.2\%$ assignments. The policies ATB^I perform inferior. We observe a decreasing solution quality with increasing interval durations. Further, anticipation improves the myopic equivalent by an equal amount per policy. This also manifests in the average improvement Δ of ATB^X compared to ATB^N and ATB^I as well as for the myopic equivalents shown in Fig. 3b. The average improvement of ATB^X over ATB^N is $\Delta(\text{ATB}^\text{X}, \text{ATB}^\text{N}) = 0.66\%$. The improvement compared to ATB^I ranges between 0.8 and 4.6% and increases with the interval durations.

Looking at the individual 108 instance settings, ATB^X provides the best results in 60 cases, ATB^N in 45, and ATB^I in only one case. In two cases, all policies provide the same results. In essence, the best results are either achieved by ATB^X following the exogenous trigger of a new request or by ATB^N following the endogenous trigger when the vehicle finished service. Planning in fixed intervals is generally inferior. In the following subsection, we therefore compare the exogenous and the endogenous trigger in detail.

5.4 Exogenous vs. Endogenous Trigger

Finally, we analyze which instance settings reinforce the benefit of the exogenous and endogenous triggers, respectively. To this end, we analyze the improvement $\Delta(\text{ATB}^\text{X}, \text{ATB}^\text{N})$ of exogenously triggered decision making ATB^X over endogenously triggered decision making ATB^N with respect to the instances' dimensions.

Figure 4 shows the improvement with respect to the *DOD* and the request distribution. The results are the average over the different numbers of customers and the service area sizes. On the x-axis, the three distributions and the three different *DOD* for every distribution are depicted. The y-axis shows the improvement. We first analyze the improvement with respect to the customer distributions. The distributions vary in the customer spread starting with widely spread customers given the uniform distribution (U) and ending with customers accumulated only in two clusters (2C). We observe a correlation of the improvement and the spread. Given widely spread customers, travel durations between customers are high and diversion is highly beneficial leading to efficient routing. If the spread is low, travel distances are low, too, and diversion does not provide significant benefits. The same behavior can be observed for the three individual service area sizes (not depicted in Fig. 4). On average, the improvement for the large service area is 0.99%, for the medium-sized service area 0.90%, and for the small service area −0.16%. Again, this can be explained by the different travel durations between the customers.

We also observe a significant impact of the *DOD*. Given a low *DOD* of 0.25, ATB^X outperforms ATB^N. For a high *DOD*, we observe an opposed behavior. This can be explained by the different information gain between the visits of two customers. Given a high *DOD*, many requests may accumulate and this information gain allows for effective assignments. If the *DOD* is low, the expected number of requests is low, too. Hence, the information gain cannot compensate the loss in routing efficiency.

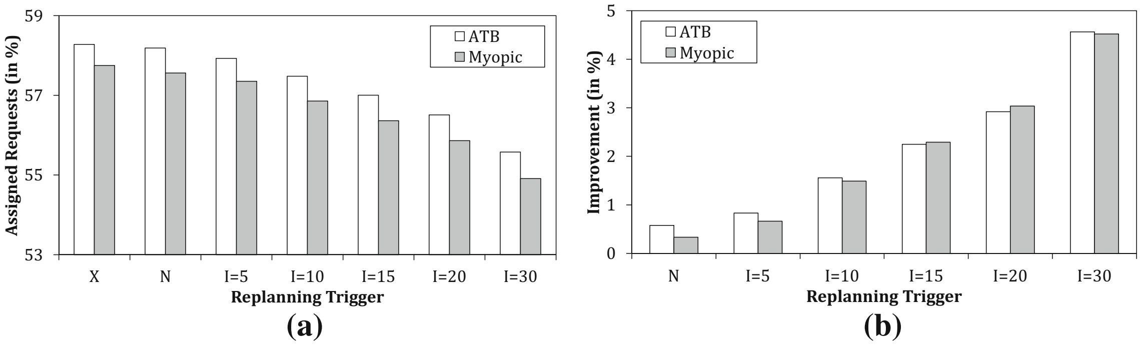

Fig. 3 Results: solution quality and improvement of ATB^X. **a** Solution quality $Q(\pi)$. **b** Improvement $\Delta(\text{ATB}^\text{X}, \pi)$

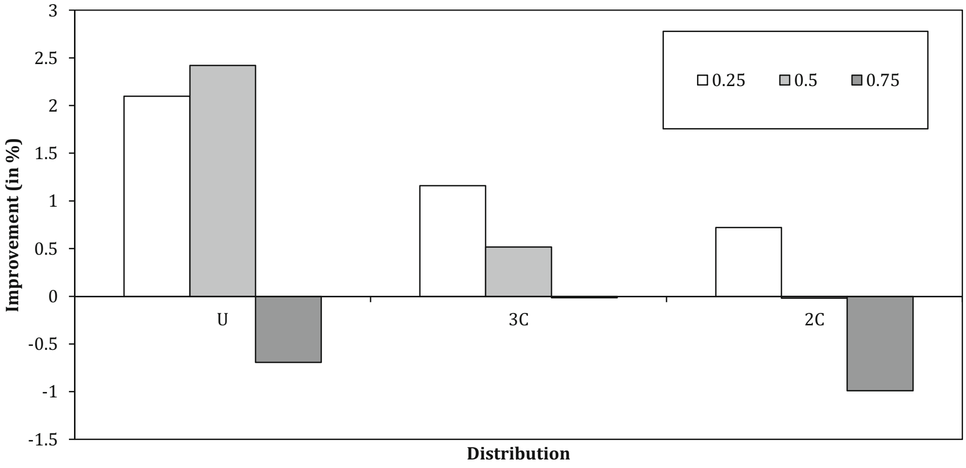

Fig. 4 Improvement per DOD and request distribution

6 Conclusion

Real-time information enables replanning at every point in time and the challenge arises to determine suitable replanning times. In this paper, we have analyzed and quantified the impact of the three most common replanning triggers for dynamic dispatching of vehicles. We have presented a BPMN model of the dispatching process and have identified customer requests and the vehicle status as main process triggers for replanning. Additionally, we have identified replanning in fixed intervals as another common practice of planning. In an extensive computational analysis, we could observe that a suitable trigger depends on the instances' characteristics. Replanning triggered by customer requests is beneficial in cases where customers are widely spread and travel durations are large as well as in cases where the number of requests is low. In these cases, diversion enabled by real-time information allows efficient routing. Replanning based on the vehicle status is beneficial given shorter travel durations and a higher request frequency. This allows an information gain and more effective assignments.

Future work may consider potential customers' and drivers' inconveniences resulting from delayed communication and diversions. Further, since replanning is increasingly outsourced to cloud providers, frequent replanning results in increasing cloud provisioning costs. Future work may consider this tradeoff between replanning costs and effective and efficient decision making. For the problem under consideration, future work may focus on identifying suitable replanning times based on current problem states. As an example, the replanning trigger may depend on the current vehicle's location. This may allow to avoid premature assignments when the vehicle traverses areas without potential requests. In these cases, diversion does not provide any benefit and decision making should be postponed until the vehicle reaches areas with more frequent requests. Further, request assignments may be made for individual requests, such as postponing assignment decisions for requests demanding service in areas the vehicle will visit only later in the day. The considered problem may be transferred to multi-vehicle settings. In these cases, the trigger selection may become more complex. Finally, the analysis could be extended by considering the impact of service times on the triggers' suitability.

Acknowledgements We appreciate the detailed and constructive comments of the external reviewers and the associate editor.

References

Angelelli E, Bianchessi N, Mansini R, Speranza MG (2009) Short term strategies for a dynamic multi-period routing problem. Transp Res Part C Emerg Technol 17(2):106–119

Bent RW, Van Hentenryck P (2004) Scenario-based planning for partially dynamic vehicle routing with stochastic customers. Oper Res 52(6):977–987

Branchini RM, Armentano VA, Løkketangen A (2009) Adaptive granular local search heuristic for a dynamic vehicle routing problem. Comput Oper Res 36(11):2955–2968

Chen X, Thomas BW, Hewitt M (2016) The technician routing problem with experience-based service times. Omega 61:49–61

Chen ZL, Xu H (2006) Dynamic column generation for dynamic vehicle routing with time windows. Transp Sci 40(1):74–88

DHL (2013) DHL improves flexibility for road freight services through mobile solution. http://www.dhl.com/content/g0/en/press/releases/releases_2013/logistics/dhl_improves_flexibility_for_road_freight_services_through_mobile_solution.html. Accessed 25 Aug 2015

Ferrucci F, Bock S (2015) A general approach for controlling vehicle en-route diversions in dynamic vehicle routing problems. Transp Res Part B Methodol 77:76–87

Gendreau M, Guertin F, Potvin JY, Taillard E (1999) Parallel tabu search for real-time vehicle routing and dispatching. Transp Sci 33(4):381–390

Ghiani G, Manni E, Quaranta A, Triki C (2009) Anticipatory algorithms for same-day courier dispatching. Transp Res Part E Logist Transp Rev 45(1):96–106

Ghiani G, Manni E, Thomas BW (2012) A comparison of anticipatory algorithms for the dynamic and stochastic traveling salesman problem. Transp Sci 46(3):374–387

Gligor DM, Holcomb M (2014) The road to supply chain agility: an RBV perspective on the role of logistics capabilities. Int J Logist Manag 25(1):160–179

Gutenschwager K, Böse J, Voß S (2003) Effiziente Prozesse im kombinierten Verkehr-ein neuer Lösungsansatz zur Disposition von Portalkränen. Logist Manag 5(1):62–73

Heilig L, Voß S, Lalla-Ruiz E (2017) port-io: an integrative mobile cloud platform for real-time inter-terminal truck routing optimization. Flex Serv Manuf. doi:10.1007/s10696-017-9280-z

Hilpert H, Kranz J, Schumann M (2013) Leveraging green IS in logistics. Bus Inf Syst Eng 5(5):315–325

Hvattum LM, Løkketangen A, Laporte G (2006) Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic. Transp Sci 40(4):421–438

Hvattum LM, Løkketangen A, Laporte G (2007) A branch-and-regret heuristic for stochastic and dynamic vehicle routing problems. Networks 49(4):330–340

Ichoua S, Gendreau M, Potvin JY (2000) Diversion issues in real-time vehicle dispatching. Transp Sci 34(4):426–438

Ichoua S, Gendreau M, Potvin JY (2006) Exploiting knowledge about future demands for real-time vehicle dispatching. Transp Sci 40(2):211–225

Lorini S, Potvin JY, Zufferey N (2011) Online vehicle routing and scheduling with dynamic travel times. Comput Oper Res 38(7):1086–1090

Lowe J, Khan AA, Bhatale B (2014) Same-day delivery: surviving and thriving in a world where instant gratification rules. Whitepaper Cognizant 20-20 Insights, Cognizant

Mulley C, Nelson JD (2009) Flexible transport services: a new market opportunity for public transport. Res Transp Econ 25(1):39–45

OMG (2013) Business process model and notation (BPMN). Tech. Rep., Open Management Group (OMG). http://www.omg.org/spec/BPMN/. Accessed 25 Aug 2015

Oyola J, Arntzen H, Woodruff DL (2016a) The stochastic vehicle routing problem, a literature review, part I: models. EURO J Transp Logist, pp 1–29

Oyola J, Arntzen H, Woodruff DL (2016b) The stochastic vehicle routing problem, a literature review, part II: solution methods. EURO J Transp Logist, pp 1–40

Powell WB (2011) Approximate dynamic programming: solving the curses of dimensionality. Wiley Series in Probability and Statistics, vol 842. Wiley, New York

Puterman ML (2014) Markov decision processes: discrete stochastic dynamic programming. Wiley, New York

Regan A, Mahmassani H, Jaillet P (1996) Dynamic decision making for commercial fleet operations using real-time information. Transp Res Rec J Transp Res Board 1537:91–97

Ritzinger U, Puchinger J, Hartl RF (2016) A survey on dynamic and stochastic vehicle routing problems. Int J Prod Res 54:215–231

Rosenkrantz DJ, Stearns RE, Lewis PM (1974) Approximate algorithms for the traveling salesperson problem. In: Ieee conference record of 15th annual symposium on switching and automata theory, IEEE, pp 33–42

Savelsbergh M, Van Woensel T (2016) City logistics: challenges and opportunities. Transp Sci 50(2):579–590

Sheridan PK, Gluck E, Guan Q, Pickles T, Balcıog B, Benhabib B et al (2013) The dynamic nearest neighbor policy for the multi-vehicle pick-up and delivery problem. Transp Res Part A Policy Pract 49:178–194

Speranza MG (2016) Trends in transportation and logistics. Europ J Oper Res **(in press)**

Thomas BW (2007) Waiting strategies for anticipating service requests from known customer locations. Transp Sci 41(3):319–331

Thomas BW, White CC III (2004) Anticipatory route selection. Transp Sci 38(4):473–487

Ulmer MW, Mattfeld DC, Köster F (2016) Budgeting time for dynamic vehicle routing with stochastic customer requests. Transp Sci **(to appear)**

Wilding R, Wagner B, Gligor DM, Holcomb MC (2012) Understanding the role of logistics capabilities in achieving supply chain agility: a systematic literature review. Supply Chain Manag 17(4):438–453