Dynamic multi-period vehicle routing: approximate value iteration based on dynamic lookup tables

Marlin W. Ulmer, Dirk C. Mattfeld, Ninja Soeffker
Technische Universität Braunschweig, Germany,
m.ulmer@tu-braunschweig.de

March 2, 2016

Abstract

We consider a dynamic multi-period vehicle routing problem with stochastic service requests. Requests occur within the days, i.e., periods and can be either accepted for same-day service or postponed to the following period. The objective is to maximize the number of same-day services over all periods with respect to limited time per period. For the single-period version of the problem, the existing anticipatory time budgeting approach (ATB) allows intra-period anticipation of future requests. ATB estimates the expected number of future same-day services regarding the point of time and the free time budget left in the period by means of approximate value iteration. We extend ATB to mATB considering free time budget of the current and following period to allow inter-period anticipation. Since the consideration of multiple periods leads to an exponential increase in the state space size, we combine approximate value iteration with a dynamic lookup table (DLT) for dynamic adaptations of the state space partitioning. Computational studies show that the DLT is mandatory to achieve inter-period anticipation. 

Keywords: multi-period dynamic vehicle routing, stochastic requests, time budget, approximate dynamic programming, approximate value iteration, dynamic lookup table
1 Introduction

Challenges for logistic service providers increase. Customers expect reasonably priced, fast, and reliable service (Ehmke 2012). In many cases, customers request services and expect service fulfillment the day of request or the day after. Service providers dynamically route service vehicles serving these requests in the course of the day (called period). Service requests are stochastically. Respective customers are generally arbitrarily distributed in the whole service region and their locations may not be known in advance. Still, service providers may be able to derive a distribution of potential customer locations, e.g., by analyzing historical data (Hvattum et al. 2006). To include new service requests in the tour, the providers have to dynamically adapt the current routing plan (Gendreau et al. 2006). Due to working hour restrictions, service providers may not be able to serve all dynamic requests the same period. In contrast to the well established single-period version of this problem, customers are not rejected, but postponed and served in the following period (Angelelli et al. 2009).

The resulting routing problem can be modeled as a multi-period dynamic vehicle routing problem with stochastic customer requests (MDRPSR). A vehicle serves customers during working hours over a finite set of periods. Every period, the vehicle starts and ends its tour in the depot. Early request customers, i.e., postponed customers from the period before have to be served and are known in advance. During each period, new requests arrive stochastically within the service area. Arriving at a customer provides a set of new requests. The dispatcher has to decide whether to accept a request for same-day service or postpone the request to the next period. The acceptances and postponements are permanent. Further, the next customer to be visited has to be selected. The objective for the MDRPSR is to find an acceptance and routing policy maximizing the expected number of same-day services over all periods.

For the MDRPSR, decisions contain both routing and acceptance and impact the current and the following periods. Efficient decisions are characterized by intra-period and inter-period anticipation. Postponing a customer in a low fre-
quented area may avoid a significant detour and allow to serve later requests in
the current period. Nevertheless, it may obstruct the routing in the next period,
resulting in less same-day services in the following periods. Hence, the decision
maker has to anticipate possible new requests in the current and following periods.

This paper presents ways to achieve intra- and inter-period anticipatory accep-
tance policies. Given a set of new requests, such policies allow for decisions to
be made about accepting or postponing a request. Anticipation is achieved by ap-
proximate value iteration (AVI), a method of approximate dynamic programming
(ADP, Powell 2011). In the decision process, each decision results in a known
post-decision state (PDS). Such a PDS consists of the point of time, the location
of the vehicle, the remaining customers in the current period, and the customers to
be served in the following period. As indicated earlier, the characteristics of a PDS
have a significant impact on the expected number of future same-day services, i.e.,
the value $V$ of a PDS. AVI approximates the value for every PDS via simulation.
Since the time to select a decision in the online execution phase is highly lim-
ited, the simulation have to be conducted offline. For this problem, the number
of possible PDSs is vast due to arbitrary customer locations. The values have to
be stored and the success of the approximation significantly depends on the num-
ber of value observations (Barto 1998, p. 193). Hence, a distinct approximation
of every PDS’s value can not be achieved. For a similar, single-period problem,
Ulmer et al. (2015d) have introduced the AVI-approach ATB representing a PDS
by a vector $(t, b)$ of point of time $t$ of the shift and the remaining free time budget
$b$. ATB approximates the value of every vector of the resulting two-dimensional
vector space. To allow inter-period anticipation, we extend ATB to mATB by
adding the current period $\delta$ and the free time budget of the next period $b_{next}$ to the
vector $(\delta, t, b, b_{next})$. The addition of this parameter exponentially increases
the considered vector space. An approximation of every vector’s value may be com-
putationally intractable. To reduce dimensionality, the vector space is partitioned
by assigning the parameters to intervals. These intervals generate a lookup table
(LT, Sutton and Barto 1998). Every axis of the LT represents a parameter. The
axes are divided in intervals. The selection of the interval length is essential for the success of AVI (George et al. 2008). To avoid unsuitable interval settings, Ulmer et al. (2015d) have developed a dynamic LT (DLT) that adapts to the problem specifics by dynamically changing the interval length of the parameters according to the approximation process.

We compare mATB with conventional ATB applied on a rolling horizon and myopic acceptance policies for instances of real-world size. Results show that the success of long-term inter-period anticipation depends on the geographical spread of possible customer locations. Given customers accumulated in clusters, inter-period anticipation significantly increases the number of same-day services while widely spread possible customer locations impede the possibility of inter-period anticipation. We further compare mATB drawing on DLTs with LTs of static interval length (SLTs).

Our contributions are twofold. For the MDRPSR, mATB achieves significant improvements in solution quality compared to state-of-the-art benchmark heuristics. The extensive analysis of the policies allows insight in the approach’s functionality. Methodologically, we show the advantages of DLTs compared to static LTs. The results of Ulmer et al. (2015d) already indicate that DLTs allow a more efficient approximation but eventually result in similar solution quality compared to SLTs. For the presented problem, we show that the adaptive state space partitioning is mandatory and that even an extensive number of approximation runs for SLTs is not able to compensate for the inferior static partitioning.

This paper is outlined as follows. In §2, we rigorously define the MDRPSR using a Markov decision process and give an overview on the related literature. In §3, we define mATB, DLT, and the benchmark heuristics. In an extensive computational study, we compare mATB and the benchmarks for a variety of real-world sized instances differing in customer distribution and number of periods in §4. We analyze approximation process, solution qualities, and acceptance behavior in §5. The paper concludes with a summary of the results and directions for future research in §6.
2 The MDRPSR

In this section, we formulate the MDRPSR and embed it in a stochastic dynamic programming setting using a Markov decision process. We further give an overview on the related literature.

2.1 Problem Formulation

Given a finite number of periods $\Delta = \{\delta = 1, \ldots, \delta_{\text{max}}\}$, a vehicle has to serve customers $C$ in the service area $A$. Each period contains a sequence of points of time $T = (0, 0 + \bar{t}, \ldots, t_{\text{max}})$ with basic time unit $\bar{t}$. In every period, the vehicle starts the tour in $t = 0$ at depot $D$ with location $l^D \in A$ and has to return regarding a time limit $t_{\text{max}}$. In the first period, a set of early request customers (ERC) $C^e$ is known and must be served. During and over the periods, new requests from the set of late request customers (LRC) $C^l$ arrive stochastically in the area. Notably, these requests are unknown before their time of request. The overall set of customers is $C = C^e \cup C^l$. Each customer $C \in C$ is defined by a vector $C = (l^c, \delta^c, t^c)$ containing the location $l^c \in A$, the request period $\delta^c \in \Delta$, and the request time $t^c \in T$. The travel time between two locations $l_1, l_2 \in A$ is defined by $d(l_1, l_2) \in T$.

When arriving at a customer, the dispatcher has to select the subset of new request to accept, thus postponing the remaining requests to the next period. The vehicle then travels to the next customer. These decisions are permanent. Postponed requests must be served in the following period. Postponements in $\delta_{\text{max}}$ are without consequences. The objective is to maximize the number of accepted requests, i.e. same-day services, over all periods. The single-period problem introduced by Ulmer et al. (2015a) can be seen as a special case of this problem with $\delta_{\text{max}} = 1$. Because of the stochastic nature of the problem, in a few cases, the initial tour length visiting the ERCs or the postponed customer of the previous period respectively may violate the time limit. For this period, these initial customers are visited and all following LRCs are postponed leading to no acceptances in this
period.

2.2 Markov Decision Process

The MDRPSR can be formulated as Markov decision process (MDP, Puterman 2014). In an MDP, a set of decision points \( k = 1, \ldots, K \) is given. Decision point \( k \) is connected to the subsequent decision point by the sequence of state \( S_k \), a decision \( x \), a post-decision state \( S^x_k \) and a stochastic transition \( \omega_k \) leading to the next decision point.

A decision point \( k \) occurs in two cases, at the beginning of each period and if the vehicle just served a customer. A state \( S_k \) for this problem is defined by the current period \( k \), the point of time \( t_k \), the vehicle’s location \( l^v_k \in A \), the set of customers \( C^i_k \) to serve in this period, and the set of customers \( C^{\delta+1}_k \) to serve in the following period. In \( \delta_{\text{max}}, C^{\delta+1}_k = \emptyset \). Further, \( S_k \) contains a set of new requests \( C^r \). In essence, a state is defined by \( S_k = (k, t_k, l^v_k, C^a_k, C^{\delta+1}_k, C^r_k) \). The initial state \( S_0 \) is defined by \( S_0 = (1, 0, D, C^a, \emptyset, \emptyset) \). The termination state \( S_K \) is defined by \( S_K = (\delta_{\text{max}}, t_{\text{max}}, D, \emptyset, \emptyset, \emptyset) \).

In every decision point \( k \), decisions \( x \) are taken about the subset \( C^a_k \) of \( C^r_k \) to accept and the next customer location to visit \( l^v_{\text{next}} \in \{l^v_k \cup L^x_k \cup \{D\} \) with \( L^x_k \in A \) the locations of \( C^{\delta,x}_k = C^a_k \cup C^o_k \). Let customer \( C^\text{next} \) be associated with the location \( l^v_{\text{next}} \). In case \( l^v_{\text{next}} = l^v_k \), the vehicle idles for one time unit, i.e., \( d(l^v_k, l^v_k) = t \). The reward \( R(S_k, x) = |C^a_k| \) of decision \( x \) is the number of accepted customers. A decision is feasible if \( C^a_k = \emptyset \) or if there exists at least one sequence \( \theta_k = (l^v_k, l^{ci_1}, \ldots, l^D) \) serving all \( C^{\delta,x}_k \) and returning to the depot \( D \) before \( t_{\text{max}} \). The overall duration \( \bar{d}(\theta_k) \) is defined by the sum of single travel times \( \bar{d}(\theta_k) = d(l^v_k, l^{ci_1}) + d(l^{ci_1}, l^{ci_2}) + \cdots + d(l^{ci_i}, l^D) \) with \( i = |C^{\delta,x}_k| \). The combination of state \( S_k \) and decision \( x \) results in a (known) PDS \( S^x_k \). The post-decision state is defined by \( S^x_k = (\delta_k, t_k, l^v_k, C^{\delta,x}_k \cup C^a_k, C^{\delta+1,x}_k) \) with \( C^{\delta+1,x}_k = C^{\delta+1}_k \cup C^r_k \setminus C^a_k \).

There are two types of transitions. The first one is the transition within the period, the other transition is inter-periodically. Within the period, the transition \( \omega_k \) from PDS \( S^x_k \) results in a state \( S_{k+1} \) with \( \delta_{k+1} = \delta_k, t_{k+1} = t_k + d(l^v_k, l^v_{\text{next}}) \),
\[ l_k = l_k^{\text{next}}, \quad C_k^{\delta} = C_k^{\delta,x} \setminus \{C^{\text{next}}\}, \quad \text{and} \quad C_{k+1}^{\delta+1} = C_{k+1}^{\delta+1,x}. \]

The inter-period transition occurs if \( S_k^x = (\delta_k, T, D, \emptyset, C_k^{\delta+1,x}) \) with \( \delta_k < \delta_{\text{max}} \). \( S_{k+1} \) is then defined by \( S_{k+1} = (\delta_{k+1} + 1, 0, C_k^{\delta+1,x}, \emptyset) \).

For an exemplary decision point, the according components state, decision, and post-decision state of the MDP for the MDRPSR are depicted in Figure 1. As seen on the left side of Figure 1, a problem-state \( S \) is defined by the period \( \delta \), the point of time \( t \), the vehicle’s position, the new requests, the customers to serve in the current period as seen on the top left, and the customers to visit in the following period as seen in the shaded service area on the bottom left. In the exemplary state in period \( \delta = 2 \) and time \( t = 20 \), three customers have to be visited in the current and two in the next period. Further, two new customers request service. Decisions
are made about the subset of requests to accept for same-day service and the next customer to serve. The immediate reward \( R(S, x) \) is the number of acceptances. A resulting post-decision problem-state \( S^x \) consists of period, time, vehicle’s position, the next customer to visit, and the set of customers to serve in the current and in the next period. Given a feasible decision \( x \), there exists at least one route that allows to visit all accepted customers and to return to the depot within the time limit. In the example depicted in Figure 1, decision \( x \) accepts request 1 and postpones request 2. Request 1 is set as the next customer to visit, illustrated by the solid line in the post-decision state \( S^x \). The dashed lines represent possible feasible tours \( \theta_k^\delta, \theta_k^{\delta+1} \) for both periods. The resulting free time budget is then \( b_k^\delta = t_{\text{max}} - t_k - \bar{d}(\theta_k^\delta) \) for period \( \delta \) and \( b_k^{\delta+1} = t_{\text{max}} - \bar{d}(\theta_k^{\delta+1}) \) respectively. After the decision’s application, the stochastic transition \( \omega \) is realized while traveling to the next customer providing a set of new requests and leading to the next decision state.

The objective for stochastic, dynamic decision problems is to achieve an optimal decision policy \( \pi^* \in \Pi \) leading to the highest expected sum of rewards (Puterman 2014). A policy \( \pi \) is a sequence of decision rules \( (X_0^\pi, X_1^\pi, \ldots, X_K^\pi) \) for every decision point \( k = 1, \ldots, K \). Each decision rule \( X_k^\pi(S_k) \) specifies the decision \( x \in X(S_k) \) to select when the process occupies state \( S_k \). An optimal policy maximizes the sum of expected rewards as stated in Equation 1.

\[
\pi^* = \arg \max_{\pi \in \Pi} \mathbb{E} \left[ \sum_{k=0}^{K} R_k(S_k, X_k^\pi(S_k))|S_0 \right] \tag{1}
\]

The optimal decision \( x_k^* \) in a specific state \( S_k \) can be derived as shown in Equation 2.

\[
x_k^* = \arg \max_{x \in X(S_k)} \left\{ R(S_k, x) + \mathbb{E} \left[ \sum_{j=k+1}^{K} R(S_j, X_j^\pi(S_j))|S_k \right] \right\} \tag{2}
\]

In decision step \( k \), the decision is selected that maximizes the immediate and expected future rewards (Bellman Equation, Puterman 2014). The expected fu-
uture rewards are also called the value $V(S^x_k)$ of post-decision state $S^x_k$. For the MDRPSR, the value of a post-decision state is the expected number of acceptances over all following decision points, i.e., the acceptances in the remaining time of the current period and the expected acceptances over the following periods.

### 2.3 Related Literature

The MDRPSR is a dynamic and stochastic vehicle routing problem (Kall and Wallace 1994). The problem setting is dynamic, because the dispatcher is allowed to adapt plans and decisions during and over the periods. It is also stochastic, because some customer requests are not known in the beginning of the period and the service provider can draw on stochastic information about future request behavior. For an extensive classification of stochastic and dynamic vehicle routing, the interested reader is referred to Ritzinger et al. (2015). In this review, we first focus on work regarding stochastic requests and then on work considering the multi-period case.

Work considering stochastic customer requests is mainly limited to single-period problems. A classification of anticipatory approaches for the single-period versions of the problem can be found in Ulmer et al. (2015b). Anticipation is generally achieved by sampling of new requests (Bent and Van Hentenryck 2004, Hvattum et al. 2006, Flatberg et al. 2007, Ghiani et al. 2009, Ulmer et al. 2015c,b), waiting strategies (Mitrović-Minić and Laporte 2004, Larsen et al. 2004, Thomas 2007), and approximate value iteration (Meisel et al. 2011, Ulmer et al. 2015a,d). Recently, Ulmer et al. (2015d) have presented ATB as the current state-of-the-art benchmark approach for the single-period problem. Single-period decision policies are mainly characterized by rejection of “inconvenient” requests with a high distance to the planned tour or far off fruitful regions with an expected high amount of future requests. The provided policies are intra-periodically anticipatory, but inter-periodically myopic.

Early work considering the planning of vehicle tours over multiple periods
was made by Cordeau et al. (1997), who describe the deterministic periodic vehicle routing problem (PVRP). Here, customers have to be visited more than once and the possible periods for a visit are known in advance. Although multiple tours are required to cover the customers’ demands over the periods, this problem will not be considered further as it is not dynamic or stochastic. For an overview on the deterministic PVRP, the interested reader is referred to Archetti et al. (2015). Multiple periods or multiple consecutive tours with stochastic requests are considered by Angelelli et al. (2007, 2009), Wen et al. (2010), Azi et al. (2012), Albareda-Sambola et al. (2014), Voccia et al. (2015). Angelelli et al. (2009) present a problem similar to the MDRPSR, except they allow rejections. Further, acceptances, postponements, and rejections are not permanent, but can be changed over time. They develop short-term priority rules based on the tour lengths and the number of requests postponed. This can be seen as a simpler version of ATB by Ulmer et al. (2015d). In the problem considered by Wen et al. (2010) and Albareda-Sambola et al. (2014), customers can only be served in the periods after the request. Since the problem is solved once every period, an integration of new requests in the current period’s tour is not feasible. While Wen et al. (2010) do not consider potential future requests in planning, Albareda-Sambola et al. (2014) integrate potential future customers in the tours of the next periods. Azi et al. (2012) and Voccia et al. (2015) consider problems not with multiple periods but multiple consecutive tours. For anticipation, they apply the multiple scenario approach by Bent and Van Hentenryck (2004) to integrate future requests in current planning. Both problems do not allow the integration of requests in the current tour but only in subsequent tours. These problems are related to the MDRPSR, since decisions about the integration of customers have to be made. Still, the tour durations are flexible and not limited by fixed period’s duration.
3 Multi-Periodical Anticipatory Time Budgeting

In this section, we introduce the approach of multi-period anticipatory time budgeting (mATB). mATB allows anticipatory subset selection for the MDRPSR. For routing, mATB draws on cheapest insertion recalled in §3.1. mATB origins from the single-period ATB (Ulmer et al. 2015d), we recall in §3.2. We then extend ATB to mATB in §3.3. Finally, we recall the concept of DLT in §3.4.

3.1 Routing

The routing is required in order to decide whether a customer request can be feasibly inserted into the current period’s tour and to determine for the vehicle where to go next. However, due to ongoing changes in the number and locations of requests, many online routing decisions have to be made throughout the shift. Hence, tours have to be achieved within short calculation time. To this end, the tours are generated by a cheapest insertion heuristic (Rosenkrantz et al. 1974), that is, customers are inserted in the tour such that the resulting detour is minimized. The previously existing tour is not changed apart from this insertion. The initial tour is created similarly. The decision about the requests that are to be inserted into the previous tour induces the decision about the next customer to be visited.

3.2 Anticipatory Time Budgeting

The main challenge of finding an optimal policy for the MDRPSR is to calculate the value $V(S_k^x)$ for a post-decision state $S_k^x$. This could be achieved by stochastic dynamic programming (SDP, Kall and Wallace 1994) recursively calculating the values considering the transition probabilities between two states. Nevertheless, for problems of real-world sizes, SDP is not applicable because of the number of states, decisions, and decision points, and additionally the inaccessible transition probabilities, all combined in the *curses of dimensionality* (Powell 2011). Hence, these values only can be approximated via simulation. Since the time for approxi-
mation is highly limited during the online execution of the routing, the simulations have to be conducted offline. ATB draws on the offline method of approximate value iteration (AVI). AVI requires a storage of the values. To this end, ATB uses an aggregation $\mathfrak{A}$ to map post-decision states $S^x_k$ to vectors of point of time $t_k$ and free time budget of the current period $b_k$: $\mathfrak{A}(S^x_k) = (t_k, b_k)$. Then, the resulting vector space $\mathfrak{V}$ is mapped to a lookup table $\mathcal{E}$ via partitioning $\mathfrak{I}$. The value $\hat{V}(\eta)$ for every entry $\eta \in \mathcal{E}$ is then approximated via simulation as described in the following. First, AVI assigns initial values $\hat{V}_0(\eta)$ to every entry $\eta \in \mathcal{E}$. These values induce a policy $\pi_0$ with respect to the Bellman Equation as depicted in Equation 3 for $i = 0$.

$$x^i_k = \arg \max_{x \in X(S_k)} \left\{ R(S_k, x) + \hat{V}_i(\mathfrak{A}(\mathfrak{I}(S^x_k))) \right\} \quad (3)$$

AVI subsequently simulates realizations $\omega_1, \ldots, \omega_m \in \Omega$. Within the simulation of realization $\omega_i$, the current policy $\pi_{i-1}$ is applied according to Equation 3. 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 Equation 4:

$$\hat{V}_i(\eta) = (1 - \alpha)\hat{V}_{i-1}(S^x_k) + \hat{V}_{\omega_i}(\eta) \quad (4)$$

Parameter $\alpha$ defines the stepsize of the approximation process. With $N(\eta) > 0$ the number of observations of $\eta$, ATB draws on $\alpha = \frac{1}{N(\eta)}$, i.e., the moving average over all observed values for $\eta$.

### 3.3 Extending ATB

Since mATB is supposed to allow anticipation beyond the current period, the aggregation has to integrate the next period. Further, the evaluation depends on the period itself. Early periods have a significantly higher value than later periods, since the value is the sum of the expected future same-day services over all remaining periods. Hence, we aggregate post-decision problem-states in mATB to vectors containing the period and the numerical attributes point of
time \( t_k \), free time budget in the current \( b_k^\delta \), and in the following period \( b_{k+1}^\delta \): 
\[ \mathcal{A}(S_k^\delta) = (\delta, t_k, b_k^\delta, b_{k+1}^\delta). \]
Compared to ATB, the dimensionality of the resulting vector space is high. To allow sufficient approximation by means of AVI, the number of periods generated by \( \mathcal{A} \) has to be reduced. We divide the periods in three phases as shown in Equation 5:

\[
\delta = \begin{cases} 
1 & \text{initial} \\
2, \ldots, \delta = \delta_{\text{max}} - 1 & \text{run} \\
\delta = \delta_{\text{max}} & \text{final}
\end{cases}
\]  

(5)

All periods are assigned to the initial phase \( \bar{\delta}_1 = \{1\} \), the run phase \( \bar{\delta}_2 = \{2, \ldots, \delta_{\text{max}} - 1\} \), or the final phase \( \bar{\delta}_3 = \{\delta_{\text{max}}\} \). The ERC-settings of the initial phase and period differ compared to the following periods. Further, in the final phase and period, the postponement of customers is without consequences. This may lead to different acceptance decisions compared to the run phase. We run AVI for \( \delta_{\text{max}} = 3 \) periods \( \Delta = (\bar{\delta}_1, \bar{\delta}_2, \bar{\delta}_3) \) and extend the achieved approximated policy for \( \delta_{\text{max}} > 3 \) by assigning the specific \( \delta \) to the according phase \( \bar{\delta} \).

### 3.4 Dynamic Lookup Table

Ulmer et al. (2015d) show that for the single-period problem, the approximation speed significantly depends on partitioning \( \mathcal{I} \). They experience a tradeoff between efficient approximation speed and effective approximation quality. To allow an efficient approximation, entries have to be observed frequently (Barto 1998). A LT of low detail is advantageous. Still, for high quality approximation, the partitioning has to allow heterogeneous states to be evaluated independently. Hence, a highly detailed LT is mandatory for high solution quality. As a result, conventional static partitioning approaches provide only suboptimal approximation. George et al. (2008) have presented a first partitioning approach addressing this tradeoff. Their weighted LT (WLT) draws on a set of partitioning of different level of detail. The value is the weighted sum of the single LTs entry-values. The weights depend on the number of observations and the value variation within an entry. Since the WLT requires substantial amounts of storage space, Ulmer et al.
(2015d) have introduced a more efficient dynamic LT. The DLT is a single LT dynamically adapting the partitioning with respect to the approximation process.

Figure 2 shows an exemplary development of the 2-dimensional DLT over the number of simulation runs (Ulmer et al. 2015d), each axis represents one dimension. As seen on the left, the DLT starts with large equidistant entries to achieve a fast first approximation. During the simulation runs, the DLT changes the partitioning for certain "interesting" entries of the LT regarding the approximation process. These entries are selected according to the number of observations and the value deviation. Preferably, frequently visited and heterogeneous problem-states are differentiated. On the right side of Figure 2, the resulting LT is shown. Intervals on the lower left part of the LT are highly separated indicating an area with both high observation frequency and value deviation. On the upper right part of the LT, the entry remains in the initial design. This part may be sparsely frequented and shows a low value deviation.

To decide whether an entry $\eta$ should be separated, the number of observations $N(\eta)$ and the deviation $\sigma(\eta)$ are compared to the overall average values $\bar{N}$ and $\bar{\sigma}$ as seen in Formula 6.

$$\frac{N(\eta) \sigma(\eta)}{N \bar{\sigma}} \geq \tau. \quad (6)$$

If the combination of both exceeds a certain threshold $\tau$, the entry occupies a high value deviation and/or is highly frequented. Therefore, it requires and allows separation. The separation of a selected entry $\eta^*$ results in a set of entries with
half the size for the parameters $t_k$ and $b_k$. In the 2-dimensional case of ATB, four new entries are generated. The number of observations for each new entry is set to $\frac{1}{4}N(\eta)$, the standard deviation to $\frac{1}{5}\sigma(\eta)$. For mATB, we apply the DLT for the parameters $t, b^h, b^{h+1}$. For this 3-dimensional DLT, the number of new entries is eight. The number of observations for each new entry is set to $\frac{1}{8}N(\eta)$, the standard deviation to $\frac{1}{8}\sigma(\eta)$.

3.5 Benchmark Heuristics

This work contributes in both solution quality for the specific problem and methodologically by means of the adaptive DLT. Hence, we require competing approaches in both areas. First, we show that mATB is able to achieve inter-period anticipation. Second, we show that the DLT is highly advantageous to static partitioning approaches. Therefore, we benchmark the DLT to SLTs of different entry sizes. Preliminary tests with the WLT by George et al. (2008) have resulted in computational intractability since the WLT draws on a set of SLTs. For this four-dimensional vector space, the number of entries exceeded the storage capacity of today’s computers.

To analyze the intra-period anticipation, we compare mATB with a myopic policy maximizing the immediate reward in every state. To show inter-period anticipation of our approach, we apply ATB of the single-period problem provided by Ulmer et al. (2015d) on a rolling horizon for every period. This approach allows intra-period anticipation, while acting inter-periodically myopic. The approaches are shown in Table 1 regarding their degree of anticipation.

3.6 Tuning

We test the algorithms for SLT with interval sizes of $I = 1, 2, 4, 8, 16$ minutes. Given $I = 4$, the intervals are for example $[0, 4), [4, 8)$, et cetera. The entry representation of the SLTs therefore differs from nearly quarter-hourly up to minute by minute consideration. DLTs start with an interval length of 16 minutes which
Table 1: Degree of Anticipation

<table>
<thead>
<tr>
<th>Intra-period</th>
<th>Inter-period</th>
<th>Myopic</th>
<th>Anticipatory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Myopic</td>
<td>myopic</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Anticipatory</td>
<td>ATB</td>
<td>mATB</td>
<td></td>
</tr>
</tbody>
</table>

can then be successively reduced up to 1 minute. Based on preliminary tests, we run mATB with DLTs for \( \tau = 2.0, \ldots, 5.0 \). For \( \tau = 2.0 \), separation occurs when an entry is slightly more frequently visited and shows a slightly higher value deviation. For \( \tau = 5.0 \), only a few distinguished entries with a high number of observations and high value deviation are separated. For ATB, Ulmer et al. (2015d) show that the best results are achieved by \( \tau_m = 1.0, 1.25, \ldots, 2.0 \). For ATB and mATB, we run 1 million approximation runs. For every parameter \( \tau \), we then run 10,000 evaluation runs and select the best policy.

4 Computational Evaluation

In this section, we define the test instances focusing on service area sizes and customer distributions with varying geographical spread of potential requests. We then analyze the achieved solution quality of the approaches mATB, ATB, and the myopic policy.

4.1 Instances

Since we want to analyze the impact of the geographical spread of customers, for the remaining parameters, we draw on the definition of Ulmer et al. (2015d). The time limit is set to \( t_{\text{max}} = 360 \) minutes. The basic time unit is \( t = 1 \). The vehicle travels with a speed of \( v = 25km/h \). The travel distances are Euclidean. The depot is located in the center of the service area. The initial expected number of
customers in δ = 1 is 100. The number of LRC in the first period is defined by the degree of dynamism (dod, Larsen et al. 2004). We examine instances with a dod = 0.75 leading to an expected number of 25 ERC in the first period and 75 expected LRC per period. In periods δ > 1, the dod depends on number of postponed customers of the previous period. The requests arrive independently over time following a Poisson distribution. We test the results for δ\textsubscript{max} = 3 and for δ\textsubscript{max} = 10 periods.

We assume that the geographical spread of potential customer locations has a significant impact on the possibility for inter-period anticipation. The lower the spread is, the less inter-period anticipation may be needed, since the vehicle has to visit the same regions regardless the period. The higher the spread is, the more may the inter-period anticipation be impeded since the requests occur arbitrarily in the service area. Hence, we draw on three customer distributions with different geographical spreads of customer locations. The locations of the customers are uniformly distributed (\(\mathcal{F}_U\)), grouped in two (\(\mathcal{F}_{2C}\)) or three clusters (\(\mathcal{F}_{3C}\)). Within the clusters, the customers are normally distributed. Further, the more expensive customers are, i.e., the more time is required to integrate a request, the higher may the potential of anticipation be. We test the approaches for a smaller service area \(\mathcal{A}_{15}\) of size 15\(\mathrm{km}\) × 15\(\mathrm{km}\) and a larger service area \(\mathcal{A}_{20}\) of
size $20km \times 20km$. An exemplary customer setting for $\mathcal{A}20$, $\mathcal{F}_{2C}$ and $\mathcal{F}_{3C}$ is shown in Figure 3. In the following, we define the distributions for $\mathcal{A}15$. For $\mathcal{A}15$, the numbers are reduced by 0.75. Given $\mathcal{F}_U$, a realization $l^c = (a_x, a_y)$ is defined as $a_x, a_y \sim U[0, 20]$. For $\mathcal{F}_{2C}$, the customers are equally distributed to each cluster. The cluster centers are located at $\mu_1 = (5, 5), \mu_2 = (15, 15)$. The standard deviation within the clusters is $\sigma = 1$. For $\mathcal{F}_{3C}$, the cluster centers are located at $\mu_1 = (5, 5), \mu_2 = (5, 15), \mu_3 = (15, 10)$. 50% of the requests are assigned to cluster two, 25% to each other cluster. The standard deviations are set to $\sigma = 1$. The area of customer requests for $\mathcal{F}_{2C}$ is highly limited, the geographical spread is small. For $\mathcal{F}_3$, the area and the geographical spread is increased by an additional cluster. Still, the customers only request in three clusters. For $\mathcal{F}_U$, customers can request in the entire service area. Hence, the geographical spread for $\mathcal{F}_U$ is significant. We assume that the benefit for inter-period anticipation is highest for $\mathcal{F}_{3C}$, reduced for $\mathcal{F}_{2C}$, and is hard to achieve for $\mathcal{F}_U$. Our computational evaluation later on confirms this assumption.

4.2 Solution Quality

For each instance setting, we run 10,000 test runs. For the defined instances, Table 2 shows the achieved solution quality for mATB, ATB, myopic, and the improvement of mATB compared to ATB in percent. The bold entries highlight the overall best solution quality. Both ATB and mATB significantly outperform the myopic policy for every instance. In the following, we compare mATB to ATB regarding service area size and customer distribution. As assumed, the more expensive customers are to insert, i.e., for $\mathcal{A}20$, the higher the benefit on inter-period anticipation. Additionally, the geographical spread of the distributions impacts the possibility of inter-period anticipation. mATB performs well for $\mathcal{F}_{3C}$ in any case and increases solution quality up to 4.7% in comparison to ATB. For $\mathcal{F}_{2C}$, mATB still outperforms ATB, but for $\mathcal{A}15$, the difference is less distinct. mATB is not able to achieve significantly better results than ATB for $\mathcal{F}_U$. For $\mathcal{A}20$, mATB is even inferior to ATB. Here, the high geographical spread does not allow long-term
inter-period anticipation.

We now analyze the impact of the number of periods. For \( F_{2C} \) and \( F_{3C} \), the solution quality with \( \delta_{\text{max}} = 3 \) and \( \delta_{\text{max}} = 10 \) only slightly differs. The differentiation between initial, run, and final periods holds. Only for \( F_U \) and \( A_{20} \), a significant decrease in solution quality can be observed. Due to the high geographical spread and the impeded anticipation, customers accumulate over the periods and the free time budget is often already consumed by the initial tour.

### 5 Analysis

In this section, we analyze how mATB is able to achieve inter-period anticipation for \( F_{2C} \) and \( F_{3C} \) but not for \( F_U \) in §5.1. Therefore, we compare the acceptance behavior of mATB to ATB within and over the periods. We further show the advantages of DLTs compared to SLTs in §5.2. We show the solution qualities in detail examining the approximation process regarding the number of approxima-
tion runs. In the analysis, we focus on $\delta_{\text{max}} = 3$ and $A_{20}$.

### 5.1 Anticipation

We first compare the acceptance behavior within and over the periods. We select $\mathcal{F}_{3C}$ because this distribution reveals the typical acceptance behavior of the policies most visibly. Figure 4 displays the acceptance behavior for the three approaches over the periods regarding the point of time. On the x-axis the point of time is shown, on the y-axis the average percentage of acceptances per point of time over all test runs. A percentage of 0.3 indicates, that in 3 out of 10 test runs, a customer was accepted for same-day services at this point of time.

We first analyze the intra-period anticipation. Therefore, we compare mATB and ATB to the myopic policy for $\delta = 1$. The myopic policy accepts more requests in the beginning of the period, but is not able to accept many requests in the second half of the period since the time budget is already consumed. ATB shows a very selective acceptance behavior in the beginning, postponing many customers. This enables ATB to accept more customers in the second half than the other approaches. The acceptance behavior of mATB can be seen as a compromise of myopic and ATB. While ATB consequently postpones inconvenient customers, mATB considers the tradeoff between rewards of the current and next period. This behavior is reflected in the acceptances of $\delta = 2, 3$. For the myopic policy and ATB, a slight decrease in acceptances can be observed over the periods. The acceptance behavior of mATB varies over the periods. Notably, for $\delta = 3$, the behavior is similar to ATB due to the fact that postponements in the final period are without consequences.

The inter-period anticipation aims on maintaining flexibility throughout the periods. As a result, mATB may result in less same-day services in the first period $\delta = 1$. Still, in contrast to ATB, mATB avoids an obstructed tour in the following period. This behavior can be observed in particular for $\mathcal{F}_{2C}$, as we show in the following analysis. The left side of Table 3 depicts the average amount of same-day services per period for $\mathcal{F}_{2C}$. As aforementioned, ATB achieves more same-
Figure 4: Acceptance Behavior over Time and Periods for $A_{20}, F_{3C}$.
Table 3: Inter-Periodical Acceptance Behavior: $A_{20}, F_{2C}$

<table>
<thead>
<tr>
<th>Period</th>
<th>Same-Day Services (in %)</th>
<th>Free Time Budget (in min)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>mATB</td>
<td>ATB</td>
</tr>
<tr>
<td>1</td>
<td>65.1</td>
<td>65.6</td>
</tr>
<tr>
<td>2</td>
<td>65.9</td>
<td>64.6</td>
</tr>
<tr>
<td>3</td>
<td>66.7</td>
<td>64.3</td>
</tr>
</tbody>
</table>

day services than mATB in $\delta = 1$. Since ATB acts inter-periodically myopic, the percentage of same-day services decreases over the periods for ATB while it even increases for mATB. This can be explained by analyzing the average initial free time budget over the periods depicted on the right side of Table 3. The free time budget is an indicator for flexibility in a period. The lower $b$ gets, the less time remains to include same-day services in the current period. Even though in $\delta = 1$ ATB postpones significantly less customers to the following periods compared to myopic, the free time budget for ATB is less. This indicates that ATB postpones mainly inconvenient customers neglecting the impact of next period’s routing. This reduces in less flexibility and overall less same-day services compared to mATB.

5.2 Partitioning

We finally analyze the impact of the partitioning DLT. Again, we focus on $A_{20}$, since the results allow a more distinct analysis. We compare the solution quality of mATB based on DLT and mATB based on SLT. Table 4 shows the results for the best tuning and the average results of the approach group, i.e., the average over DLTs with different $\tau$ and SLTs with different $I$. A high discrepancy between best and average results indicates a high deviation in solution quality regarding the interval sizes $I$ or parameter $\tau$ respectively. Further, we depict the improvement of DLT compared to SLT. For every instance, partitioning DLT outperforms SLT significantly. Even the best SLTs achieve lower solution qualities than the average
Table 4: Solution Quality: Same-Day Services (in %), DLT vs. SLT (in %)

<table>
<thead>
<tr>
<th>Distribution</th>
<th>$\mathcal{F}_U$</th>
<th>$\mathcal{F}_{2C}$</th>
<th>$\mathcal{F}_{3C}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best Average</td>
<td>Best Average</td>
<td>Best Average</td>
</tr>
<tr>
<td>$\delta_{\text{max}} = 3$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLT</td>
<td>23.8 22.0</td>
<td>66.8 65.2</td>
<td>61.0 60.4</td>
</tr>
<tr>
<td>SLT</td>
<td>23.0 19.9</td>
<td>64.2 61.9</td>
<td>57.2 55.4</td>
</tr>
<tr>
<td>DLT vs. SLT</td>
<td>3.5 10.6</td>
<td>4.0 5.3</td>
<td>6.6 9.0</td>
</tr>
<tr>
<td>$\delta_{\text{max}} = 10$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLT</td>
<td>7.8 7.2</td>
<td>66.4 64.5</td>
<td>59.9 56.5</td>
</tr>
<tr>
<td>SLT</td>
<td>7.4 6.3</td>
<td>64.2 61.2</td>
<td>56.7 53.9</td>
</tr>
<tr>
<td>DLT vs. SLT</td>
<td>5.4 14.3</td>
<td>3.4 5.4</td>
<td>5.6 4.8</td>
</tr>
</tbody>
</table>

DLTs. The average improvement of the best DLT to the best SLT is 4.8%, for the average DLT to the average SLT even 8.2%. Notably, mATB based on SLT is not able to outperform the results of ATB depicted earlier in Table 2 for any instance. To achieve inter-period anticipation, a DLT-partitioning is mandatory.

The advantages of DLTs additionally manifest in the approximation process. To examine the approximation behavior of the approaches, in Figure 5, we depict the development of the best and average solution quality over the 1 million approximation runs, given $\mathcal{F}_{2C}$-distribution and $\delta_{\text{max}} = 3$. While DLT achieves a high quality solution fast, the SLT approximation requires a higher amount of approximation runs. Notably is the high difference in solution quality between the best and average SLT-approach. This indicates that the success of SLT significantly depends on the a priori interval size selection. For DLT, the gap between average and best value is small. This allows the assumption that DLT is a generic approach and only marginally depends on parameter $\tau$.

6 Conclusion and Outlook

In many cases, service providers have to decide whether to serve or postpone a customer request. To achieve a high number of same-day services, the anticipa-
tion of future customer requests is mandatory. In this paper, we have presented the MDRPSR, a multi-period vehicle routing problem with stochastic service requests. To achieve inter-period anticipatory acceptance policies, we have extended the state-of-the-art heuristic ATB by Ulmer et al. (2015d) to mATB. mATB estimates future rewards over the periods by means of approximate value iteration based on lookup table-partitionings. We have compared static and dynamic partitioning approaches. Experiments show that adaptive DLTs provide high quality solutions and are mandatory for intra-period and inter-period anticipation for the MDRPSR.

In future research, the AVI-approaches could be applied to instances based on real world data. Further, the problem setting could be extended including multiple vehicles and depots or simultaneous pickup and deliveries, e.g., in the field of same-day delivery. Regarding the AVI, addition of spatial information like vehicle or customer locations to the PDS-representation might be promising. Finally, the DLT-approach could be modified by allowing to merge intervals and to add or remove parameters.
References


