Anticipatory Optimization for a Dynamic Multi-Period Routing Problem with Stochastic Customer Requests

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

May 15, 2015

Abstract

We consider a multi-period routing problem. A vehicle has to serve customers for a number of periods (e.g., days), considering a time limit within the period. Each period, known early request customers (ERCs) have to be served. During the period, late request customers (LRCs) request service. The LRCs can be confirmed or postponed to the next period, where they become ERCs. Objective is to maximize the overall number of confirmations. For anticipation of future requests, we apply value function approximation (VFA) based on lookup tables (LTs). VFA evaluates states and actions regarding the expected number of future confirmations. As indicators for the expected value, we use temporal parameters time and slack. We compare classical static LTs with dynamic LTs adapting their structure during the approximation process to the problem specifics. For the presented problem, dynamic LTs outperform all static LTs and allow efficient and effective decision making within and over the periods.

keywords: multi-period vehicle routing, stochastic customer requests, value function approximation, approximate dynamic programming, dynamic lookup table
1 Introduction.

In last mile delivery, challenges for logistic service providers increase. Customers expect reasonably priced, fast and reliable service (Ehmke 2012). Due to the continuing increase in e-commerce sales (ATKearny 2013), the number of shipped parcels has grown significantly in recent years resulting in a high amount of requests for parcel pickup (Hvattum et al. 2006). Delivery vehicles follow static routes defined by service network design (Crainic 2000). In contrast to delivery vehicles, pickup vehicles are scheduled dynamically serving pickup-requests occurring in the course of the day (Lin et al. 2010). A significant subset of these pickup-requests is stochastic and not known in the beginning of the service period. The respective customers are generally arbitrarily distributed in the whole service region and their locations may not be known in advance. To include new pickup-requests in the tour, the provider has to dynamically adapt the current routing plan (Gendreau et al. 2006). Due to working hour restrictions, service provider may not be able to serve all dynamic requests the same day. In contrast to the well established single-period version of this problem, customers are not rejected, but postponed and served the following day (Angelelli et al. 2009). Hence, decisions contain both routing and confirmations and impact the current and the following service period. Therefore, anticipation of future requests into current decision making is mandatory (Meisel 2011). Efficient decisions are characterized by intraperiodical and interperiodical anticipation. Postponing a customer in a low frequented area may avoid a significant detour and allow to confirm later requests in the current period. Nevertheless, it may obstruct the routing in the next period, resulting in less confirmations. So, the decision maker has to anticipate possible new requests in the current and following period. For anticipation of future requests, service providers can derive probabilities of customer requests for certain regions of the service area and certain times by analyzing typical customer behavior (Dennis 2011) and historical data (Hvattum et al. 2006).

In this paper, we study a multi-period dynamic vehicle routing problem with stochastic customer requests. We assume that a non-capacitated vehicle collects parcels during working hours over a finite set of periods. Every period, the vehicle starts and ends its tour in the depot. Early request customers (postponed customers from the period before) have to be served and are known in advance. During each service 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 permanently accept or postpone new requests and which customer to serve next. Our objective is to maximize the number of confirmed dynamic requests over all periods considering the given time limit reflecting the driver’s working hours.

This paper aims on providing intra- and interperiodical anticipatory con-
firmation policies. Given a set of new requests, such policies allow for decisions to be made about confirming or postponing a request. In the decision process, each decision results in a known problem-state. Such a state consists of the point of time, the locations of the vehicle, the remaining customers in the current period, and the ERCs in the following period. These problem-states have an significant impact on the number of future confirmations. To consider this impact, we apply a value function approximation (VFA, Powell 2007). Policies drawing on VFA consider both the immediate and expected future confirmations. The immediate number depends directly on the applied decision. The expected number of future confirmations is approximated for every problem-state. Due to the multiplicity of possible problem-states, a distinct approximation of every state value is not achievable. Therefore, we assign a group of problem-states to a simplified VFA-post decision state (PDS) of lower dimensionality. A VFA-PDS consists of temporal key parameters time, slack in the current, and slack in the following period. To reduce dimensionality, the parameter realizations are assigned to intervals. These intervals generate the VFA-PDS space. The VFA-PDS space can be described as a lookup table (LT, (Sutton and Barto 1998)). Every axis of the LT represents a parameter. The axes are divided in equidistant intervals. The number of different entries in the table and respectively, the size of the VFA-PDS space is inversely proportional to the interval length. The selection of the interval length is essential for the success of VFA (George et al. 2008). To avoid unsuitable interval settings, Ulmer et al. (2014) 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.

For the presented problem, we compare DLTs with LTs of static interval length and greedy confirmation policies for instances of real-world size. Even though all LT-approaches allow anticipation and increase the number of confirmed customers significantly, solution quality of static LTs (SLTs) is limited. Additionally, we experience a high variance in solution quality regarding different interval lengths. In contrast to SLT, DLTs prove as generic outperforming SLTs by up to 5.7% allowing both intraperiodical and interperiodical anticipation.

This paper is outlined as follows. In §2, we present and discuss the related literature focusing on single- and multi-periodical vehicle routing problems with stochastic customer requests. In §3, we define the vehicle routing problem using a Markov Decision Process. In §4, we recall the concept of value function approximation based on dynamic LTs. The algorithms for the presented problem are defined in §5. For a variety of real-world sized instances differing in customer distribution and number of periods, we apply the LT-approaches and analyze approximation process, solution qualities, and confirmation behavior in §6. The paper concludes with a summary of the results and directions for future research in §7.
2 Literature Review

We consider a dynamic and stochastic vehicle routing problem (Kall and Wallace 1994). Dynamic, because the dispatcher is allowed to adapt plans and decisions during and over the periods. Stochastic, because some customer requests are not known in the beginning of the period. For an extensive classification of stochastic and dynamic vehicle routing, the interested reader is referred to Pillac et al. (2013).

Work considering stochastic customer requests is mainly limited to single-periodical problems. A classification of anticipatory approaches for the single-period versions of the problem can be found in Ulmer et al. (2014). Anticipation is generally achieved by sampling of new requests (Bent and Van Hentenryck 2003, Bent and Van Hentenryck 2004, Hvattum et al. 2006, Flatberg et al. 2007, Ghiani et al. 2009, Ulmer et al. 2015b), waiting strategies (Larsen et al. 2004, Thomas 2007), and value function approximation (Meisel et al. 2011, Ulmer et al. 2014, Ulmer et al. 2015a). Single-period solutions 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 intraperiodically anticipatory, but interperiodically myopic.

The first approach extending the problem to a multi-periodical case was made by Angelelli et al. (2009). They consider a problem, where a set of vehicles has to serve stochastic pickup-requests over a number of periods. Some customers can be postponed or even rejected, if the integration in a feasible tour is not possible. Confirmations, postponements and rejections are not permanent, but can be changed over time. Objective is to minimize both the number of rejected requests and tour length per day. Angelelli et al. (2009) develop short-term optimization policies, applied hourly, considering tour lengths and the number of requests postponed. They show that including the next days routing into decision making allow interperiodical anticipation decreasing the number of rejections significantly. In 2010, Angelelli et al. (2010) extended the problem setting. Amongst others, customer confirmations have to be immediate and permanent providing more customer oriented decisions on the expense of routing costs. Further, they reduce the hourly gap between decision points to gaps of 30 seconds to allow faster and more effective decision making. The routing problem defined in section §3 follows this motivation extending the single-period version introduced by Ulmer et al. (2015a).

3 A Multi-Periodical Routing Problem

In this section, we formulate the problem and embed it in a stochastic dynamic programming setting using a Markov Decision Process.
3.1 Problem Formulation

Given a finite number of periods $\delta = 1, \ldots, \delta_{\text{max}}$, a vehicle has to serve customers in the Euclidean plane. In every period, it starts the tour in a depot and has to return regarding a time limit $t_{\text{max}}$. In the beginning of each period, a set of early request customers is known and must be served. During a period, new requests arrive stochastically in the plane. When arriving at a customer, the dispatcher has to select the subset of new request to accept postponing the remaining requests. Then, the vehicle travels to the next customer. These decisions are permanent. Postponed requests are the ERC of the following period. Postponements in $\delta_{\text{max}}$ are without consequences. Objective is to maximize the number of confirmed 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 in the beginning of a period may violate the time limit. For this period, all ERCs are visited and all following LRCs are postponed leading to no rewards in this period.

3.2 Markov Decision Process

This stochastic and dynamic problem can be formulated as Markov Decision Process (MDP, Bellman 1957b). The according components of the MDP for the given problem are exemplarily 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 vehicles 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 $x \in X$ are made about the subset of requests to confirm and the next customer to visit. The immediate reward $R(x)$ is the number of confirmations. Given a feasible decision, it exists at least one route that allow to visit all confirmed customers and to return to the depot within the time limit. In the example depicted in Figure 1, decision $x_1$ confirms 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 problem-state $p$ on the right side. The dashed lines represent possible feasible routes for both periods. As seen on the right side of Figure 1, a post decision problem-state consists of period, time, vehicles position, the next customer to visit, and the set of customers to serve in the current and in the next period. For this exemplary problem-state, adding request 1 or request 2 would consume an equal amount of driving time in the current period leading to the same immediate reward $R = 1$. Nevertheless, postponing
request 2 leads to a significantly shorter route in period $\delta = 3$. So, the expected number of future confirmations might be higher confirming request 1. After the decisions application, the stochastic transition $\omega$ is realized while traveling to the next customer providing a set of new requests.

4 Anticipatory Optimization

In this section, we recall the concept of value function approximation. Further, we explicitly illustrate the functionality of dynamic lookup-tables.

4.1 Value Function Approximation

The Objective for stochastic, dynamic problems is to achieve an optimal decision policy $\pi \in \Pi$ leading to the highest expected number of rewards (Goodson et al. 2014). A policy 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 to select when the process occupies state $s_k$. An optimal policy maximizes then sum of expected rewards as stated in Equation 1.

$$
\max_{\pi \in \Pi} E \left[ \sum_{k=0}^{K} R_k(X_k^\pi(s_k)) | s_0 \right] \tag{1}
$$
The optimal decision $x^*_{k}$ in a specific state $s_k$ can be derived as shown in Equation 2.

$$x^*_{k} = \arg \max_{x \in X(s_k)} \left\{ R(x) + \mathbb{E} \left[ \sum_{j=k+1}^{K} R(X^*_j(s_j)) | s_k \right] \right\}$$ (2)

In decision step $k$, the decision is selected maximizing the immediate and expected future rewards (Bellman Equation, Bellman 1957a).

The main challenge of finding an optimal policy is to calculate the expected future rewards. This could be achieved by stochastic dynamic programming (SDP, Kall and Wallace 1994) recursively calculating the expected 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 inaccessible transition probabilities all combined in the *curses of dimensionality* (Powell 2007).

An approach to approximate an optimal policy is value function approximation. VFA generally lowers dimensionalities to sufficiently estimate the expected values. Therefore, VFA reduces the number of states by aggregation, the number of decisions by decomposition, the number of decision points and transition probabilities by sampling. Post decision problem-states are aggregated to a VFA-PDS vector of key parameters. The expected future rewards are represented by the VFA-PDS’s value. By repeatedly sampling of problem trajectories, the values are approximated. Further, many approaches reduce the decision space by decomposition to limit the number of possible decisions and to allow efficient approximation.

4.2 Dynamic Lookup Table

The VFA-PDS values are stored in a multidimensional lookup table. Each axis of the LT represents a key parameter. Each axis representing a numerical key parameter is divided in intervals. The interval lengths define the LT size and have a huge impact on the success of VFA (Sutton and Barto 1998). A large LT with small interval lengths allows a detailed consideration of different problem-states. This differentiation may increase the solution quality. A small LT with large intervals results in a frequent visit of the LT-entries and therefore, a fast and reliable approximation. To achieve both a fast approximation and a detailed consideration of different problem-states, Ulmer et al. (2014) introduced a dynamic LT (DLT). This generic approach adapts the interval length to the problem specifics.

Figure 2 shows an exemplarily development of a 2-dimensional LT over the number of trajectories. As seen on the left, the DLT starts with large intervals to achieve a fast first approximation. During the trajectory runs, the DLT changes the interval lengths in certain ”interesting” entries of the
LT regarding the approximation process. These entries are selected considering 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 disaggregated indicating an ”interesting” area with 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 has a low value deviation.

To decide, if an entry $p$ is disaggregated, the number of observations $N(p)$ and the deviation $\sigma(p)$ is compared to the overall average values $\bar{N}$ and $\bar{\sigma}$ as seen in Formula 3.

$$\frac{N(p) \sigma(p)}{N \bar{\sigma}} \geq \tau.$$  \hspace{1cm} (3)

If the combination of both exceed a certain threshold $\tau$, the entry occupies a high value deviation and is highly frequented. Therefore, it requires and allows disaggregation.

5 VFA for the multi-periodical vehicle routing problem

In this section, we define and tune the VFA-approaches for the problem introduced in section §3. Further, we generalize the approach to allow decision making for an arbitrary number of periods.

5.1 State Space Representation

Temporal attributes like time and slack have proven as reliable indicators to estimate a states value (Angelelli et al. 2009, Meisel et al. 2011, Ulmer et al. 2014, Ulmer et al. 2015a). To apply VFA to the given problem, we define VFA-PDD using only temporal attributes. We aggregate post decision problem-states to vectors containing the period and the numerical attributes point of time, slack in the current, and in the following period.
So, the problem-PDS depicted in Figure 1 is assigned to vector \( (\delta = 2, t = 20, s_1 = 120, s_2 = 180) \), assuming the slack in period \( \delta = 2 \) is 120 minutes, and in \( \delta = 3 \), 180 minutes.

5.2 Routing and Confirmations Policies

The VFA-algorithms settings follow Ulmer et al. (2014). To reduce the number of possible decisions, we use a decomposition of the problem into routing and confirmation policies. For routing, we use cheapest insertion (Rosenkrantz et al. 1974). Cheapest insertion proofs appropriate for the given problem, because it includes new customers maintaining the tour’s main structure and the customers sequence. This may enable the dispatcher to communicate an estimate arrival time to the customers. As a result, we are looking to achieve an anticipatory confirmation policy.

We design confirmation policies using VFA based on static and dynamic LTs disaggregating the numerical parameters point of time, slack in the current and the following period. We run the VFA for \( \delta_{\text{max}} = 3 \) periods, extending the achieved approximated value functions to a higher number of periods by dividing the periods in three phases as shown in Equation 4. The initial phase \( \delta = 1 \), the run phase \( \delta = 2, \ldots, \delta_{\text{max}} - 1 \) and the final phase \( \delta = \delta_{\text{max}} \). The ERC-settings of the initial phase and period differs compared to the following periods. Further, in the final phase and period, the postponement of customers is without consequences. This may lead to different confirmation decisions compared to the run phase.

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

(4)

We test the algorithms for static interval sizes \( I = 1, 2, 4, 8, 16 \) minutes. SLTs entry representation differs from nearly quarter-hourly up to minute by minute consideration. DLTs start with interval length of 16 minutes and are successively reduced up to 1 minute. We test DLTs for \( \tau = 2.0, \ldots, 5.0 \). For \( \tau = 2.0 \), disaggregation 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 disaggregated. For every VFA-approach, we run 1 million approximation runs.

As benchmark, we first compare the LT-approaches with a myopic confirmation policy (Greedy). To show interperiodical anticipation of our approach, we additionally apply the VFA DLT\(_m\) of the single-period problem provided by Ulmer et al. (2014). This approach allows intraperiodical anticipation, while acting interperiodically myopic. For DLT\(_m\), best results are achieved by \( \tau_m = 1.0, 1.25, \ldots, 2.0 \) (Ulmer et al. 2014). The approaches are shown in Table 1 regarding their degree of anticipation.
Table 1: Degree of Anticipation

<table>
<thead>
<tr>
<th></th>
<th>Interperiod</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intraperiod</td>
<td>Myopic</td>
</tr>
<tr>
<td></td>
<td>Anticipatory</td>
</tr>
<tr>
<td>Myopic</td>
<td>Greedy</td>
</tr>
<tr>
<td>Anticipatory</td>
<td>DLT&lt;sub&gt;m&lt;/sub&gt;</td>
</tr>
</tbody>
</table>

6 Computational Evaluation

In this section, we define the test instances oriented on real world settings. We run VFA and compare the solution quality of the different LT-approaches. We especially examine the approaches impact on routing and the confirmation behavior within and over the periods. For $\delta_{\text{max}} = 3$, we show the solution qualities in detail examining the approximation process regarding the number of approximation runs. For the achieved confirmation policies, we exemplarily show the intraperiodical and interperiodical impacts to routing and confirmation compared to the myopic approaches.

6.1 Instances

The instances are provided by Ulmer et al. (2014) and are extended to multi-periodical settings. The time limit is set to $t_{\text{max}} = 360$ minutes. We test the approaches for a $20km \times 20km$ service area. The travel distances are Euclidean. The vehicle travels with a speed of $v = 25km/h$. The depot is located in the center of the area. The initial expected number of customers in $\delta = 1$ is 100. The number of LRC in the first period is defined by the degree of dynamism ($dod$, Larsen et al. 2002). We examine instances with a $dod = 0.75$ leading to 75 expected LRC per period. (In periods $\delta > 1$, the $dod$ depends on number of postponed customers of the previous period.) The requests arrive independently over time following a poisson distribution. The locations of the customers are uniformly distributed (U), grouped in two (2C) or three (3C) clusters. Within the clusters, the customers are normally distributed. An exemplary customer setting for 2C and 3C is shown in Figure 3. We test the results for $\delta_{\text{max}} = 3$ and for $\delta_{\text{max}} = 10$ periods.

6.2 Solution Quality

For each instance settings, we run 10,000 test runs. For the defined instances, Table 2 shows the average and best solution qualities regarding the parameter settings of $I$ and $\tau$ for DLTs, SLTs, and the greedy confirmation policy (Greedy). The bold entries highlight the overall best solution quality. For all instances, DLTs (respective, DLT<sub>m</sub>) achieve the highest number of confirmations, outperforming SLT and Greedy.
A high discrepancy between best and average values indicates a high deviation in solution quality regarding the interval sizes $I$, respectively the disaggregation parameter $\tau$. While for the DLTs, the discrepancy is low, SLTs solution qualities show a high variance depending on interval size $I$. Further, even the best static lookup-table achieves lower solution quality than the average DLT-approach. As predicted in section 4.2, for small intervals in the SLT, some entries are sparsely visited providing unreliable values and decreasing solution quality (missing reliability). For large intervals, heterogeneous states are grouped to one distorted value (missing accuracy). This reveals the advantages of a lookup-table dynamically adapting to problem and instance settings. The results show, DLTs allow both reliability and accuracy leading to significantly better and different confirmation policies than SLTs.

In cases, where DLT$_m$ provides better solutions than DLT, interperiodical anticipation might be difficult. For uniformly distributed customers (U), DLT performs significantly worse than DLT$_m$. The reason lies in the unpredictable customer locations and is examined in more detail in section

Table 2: Solution Quality: Confirmations (in %)

<table>
<thead>
<tr>
<th>Distribution</th>
<th>U</th>
<th>2C</th>
<th>3C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Best</td>
<td>Average</td>
<td>Best</td>
</tr>
<tr>
<td>$\delta_{\text{max}} = 3$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLT</td>
<td>23.8</td>
<td>22.0</td>
<td>66.8</td>
</tr>
<tr>
<td>SLT</td>
<td>23.0</td>
<td>19.9</td>
<td>64.2</td>
</tr>
<tr>
<td>DLT$_m$</td>
<td><strong>32.1</strong></td>
<td><strong>31.7</strong></td>
<td>64.6</td>
</tr>
<tr>
<td>Greedy</td>
<td>18.4</td>
<td>18.4</td>
<td>61.3</td>
</tr>
<tr>
<td>$\delta_{\text{max}} = 10$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DLT</td>
<td>7.8</td>
<td>7.2</td>
<td><strong>66.4</strong></td>
</tr>
<tr>
<td>SLT</td>
<td>7.4</td>
<td>6.3</td>
<td>64.2</td>
</tr>
<tr>
<td>DLT$_m$</td>
<td><strong>11.9</strong></td>
<td><strong>11.3</strong></td>
<td>64.3</td>
</tr>
<tr>
<td>Greedy</td>
<td>5.9</td>
<td>5.9</td>
<td>61.0</td>
</tr>
</tbody>
</table>
§6.4. For 2C and 3C customer distribution, DLT outperforms the other approaches indicating the merit of interperiodical anticipation.

Comparing the number of periods, for the clustered distributions, only a slight decrease in solution quality from $\delta_{\text{max}} = 3$ to $\delta_{\text{max}} = 10$ can be observed. This indicates that ERC-settings in the run phase are nearly independent of the current period and the value function approximation achieved for $\delta = 2$ holds for every period of the run phase. Therefore, the assumption of the three phases is valid. For the uniformly distributed customers, the solution quality drops significantly. As shown in section §6.4, the number of postponements increases over the periods leading to a more and more obstructed ERC-setting.

The advantages of DLTs additionally manifest in the approximation process. To examine the approximation behavior of the approaches, in Figure 4, we depict the development of the best and average solution quality over the $I = 1$ million approximation runs, given 2C-distribution and $\delta_{\text{max}} = 3$. The best approaches for this instance are SLT($I = 4$) and DLT($\tau = 4.0$). While DLT achieves fast high quality solution, SLT approximation requires a higher amount of approximation runs. Striking 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 depend of parameter $\tau$. 

Figure 4: Approximation Process for DLT and SLT given 2C Customer Distribution
6.3 Intraperiodical

To examine the intraperiodical confirmation behavior, Figure 5 displays the confirmation behavior for period $\delta = 1$ regarding the point of time for Greedy and DLT, for 2C. The confirmation behavior matches the single-period structure depicted in Ulmer et al. (2014). Greedy confirms more customers in the beginning of the period, but is not able to confirm many customers in the second half of the period. DLT confirms a slightly lower amount in the beginning allowing to achieve significantly more confirmations in the remaining time of the period compared to Greedy. $\text{DLT}_m$ acts very selective in the beginning and is able to confirm more customer in the second half than the other approaches. The many postponements in the beginning lead to a more obstructed tour in the following period as shown in section §6.3.

6.4 Interperiodical

In the following, we want to examine the interperiodical confirmation behavior. Therefore, we exemplarily select Greedy, $\text{DLT}(\tau = 3.0)$ and $\text{DLT}_m(\tau_m = 1.5)$. We show the impact of customer distribution on interperiodical anticipation comparing the results for 2C and uniformly distribution. Table 3 shows the average amount of confirmation regarding the periods for the 2C customer distribution. As expected, $\text{DLT}_m$ allows the highest number of confirmations for all approaches in period $\delta = 1$. Nevertheless, like Greedy, solution quality decreases over the periods, while DLT maintains and even improves solution quality. The decrease of $\text{DLT}_m$ results from the avoidance of ”inconvenient” customers in $\delta = 1$. So, $\text{DLT}_m$ acts interperiodical greedy. This manifests in the development of the average amount of slack over the
periods, additionally depicted in Table 3. While slack for DLT even raises in $\delta = 3$, slack for DLT$_m$ and Greedy continuously drops leading to less time budgets in later periods.

<table>
<thead>
<tr>
<th>Period</th>
<th>DLT</th>
<th>DLT$_m$</th>
<th>Greedy</th>
<th>DLT</th>
<th>DLT$_m$</th>
<th>Greedy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>65.1</td>
<td>65.6</td>
<td>61.4</td>
<td>198.3</td>
<td>198.3</td>
<td>198.3</td>
</tr>
<tr>
<td>2</td>
<td>65.9</td>
<td>64.6</td>
<td>60.3</td>
<td>190.6</td>
<td>186.4</td>
<td>188.1</td>
</tr>
<tr>
<td>3</td>
<td>66.7</td>
<td>64.3</td>
<td>59.8</td>
<td>192.1</td>
<td>184.3</td>
<td>185.6</td>
</tr>
</tbody>
</table>

The solution quality over the periods for uniformly distributed customers in depicted in Table 4. For this instance setting, predictions of future request locations are hardly possible. In many cases, new requests can not be integrated in the current tour and additionally obstruct the preplanned tour of the following period. This often results in a violation of the time limit. The percentage of time limit violations over the periods is shown in Table 4. Even though DLT is not able to avoid every violation, it allows a significant reduction of violations up to 19.8% compared to Greedy. Nevertheless, acting only short-term anticipatory like DLT$_m$ results in the best solution qualities. In essence, for the uniformly distributed customer requests, interperiodical anticipation is not possible.

7 Conclusion and Outlook

In many cases, service providers have to decide, whether to serve or postpone a customer request. To select effective decisions, the anticipation of future customer requests is mandatory. In this paper, we presented a multiperiodic vehicle routing problem with stochastic customer requests. For achieving anticipatory confirmation policies, we applied value function approximation based on lookup tables. We compared LTs with static and dy-

<table>
<thead>
<tr>
<th>Period</th>
<th>DLT</th>
<th>DLT$_m$</th>
<th>Greedy</th>
<th>Violations (in %)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>38.6</td>
<td>45.9</td>
<td>35.2</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>20.0</td>
<td>32.0</td>
<td>14.8</td>
<td>11.4</td>
</tr>
<tr>
<td>3</td>
<td>8.4</td>
<td>18.6</td>
<td>4.0</td>
<td>58.9</td>
</tr>
</tbody>
</table>
namic interval length. Experiments show that DLTs generically providing high quality solution allowing intraperiodical and interperiodical anticipation of future customer request.

In future research, the VFA-approaches could be applied to an extended set of instances varying in dod, number of periods, and service area size or basing on real world data. Further, the problem setting could be extended including multiple vehicles and depots or simultaneous pickup and deliveries. Regarding the VFA, addition of spatial information 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


Ulmer, Marlin W., Dirk C. Mattfeld, Felix Köster. 2014. Anticipation of stochastic