European Journal of Operational Research 225 (2013) 130–141

Contents lists available at SciVerse ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Innovative Applications of O.R.

A pro-active real-time control approach for dynamic vehicle routing problems dealing with the delivery of urgent goods

Francesco Ferrucci [a], Stefan Bock [a,*], Michel Gendreau [b]

[a] Institute of Business Computing and Operations Research, University of Wuppertal, Gaussstrasse 20, 42097 Wuppertal, Germany
[b] CIRRELT and MAGI, École Polytechnique de Montréal, C.P. 6079, Succursale Centre-Ville, Montreal QC, Canada H3C 3A7

ARTICLE INFO

Article history:
Received 12 October 2011
Accepted 8 September 2012
Available online 21 September 2012

Keywords:
Dynamic vehicle routing
Real-time control
Request forecasting
Past request information

ABSTRACT

This paper proposes a new pro-active real-time control approach for dynamic vehicle routing problems in which the urgent delivery of goods is of utmost importance. Without assuming any distribution, stochastic knowledge about future requests is generated using past request information. The generated knowledge is integrated into the transportation process, which is controlled by a Tabu Search algorithm, in order to actively guide vehicles to request-likely areas before requests arrive there. By analyzing the results attained for various test settings, we identify structural diversity as a crucial criterion for classifying the quality of stochastic knowledge attainable from past request information. This criterion provides a promising starting point for assessing the quality of past request information in order to efficiently use the derived stochastic knowledge in real-time control approaches. We prove the efficiency of our approach by a direct comparison with a deterministic approach on test scenarios with varying structural diversity. Thanks to the proposed classification of structural diversity, differences in results obtained among the tested scenarios become explainable.

1. Introduction

The vehicle routing problem (VRP) is a well-studied problem in literature (for a comprehensive overview, see Toth and Vigo (2002) and Golden et al. (2008)) that has been extended to the dynamic VRP (DVRP, Psaraftis, 1988; Ghiani et al., 2003; Eksioglu et al., 2009). In the DVRP, only incomplete information is known in advance and further information, e.g., new customer requests (or in short, requests), is revealed as time unfolds (see Larsen et al., 2002, 2007). Due to these dynamic changes, the tour plan (or in short, the plan) is continuously adapted during its execution.

Most DVRP approaches focus on applications where it is sufficient that a request is serviced *within* an assigned time window, but not *when* it is serviced in its time window. However, there are numerous real-world transportation processes in which the urgent delivery of goods has the highest priority. Note that this urgency often arises from failures in an earlier delivery. In particular, recent research in marketing has revealed that an adequate handling of customers' complaints about service quality is crucial in order to maintain customer loyalty (see Brady and Cronin Jr, 2001; Pollack, 2009). Specifically, for companies delivering such goods, providing a high service quality is considered as an essential factor for successfully maintaining and expanding their market positions (see Parasuraman et al., 1985; Zeithaml et al., 1996). In this paper we introduce a variant of the DVRP with the objective of *minimizing customer inconvenience* which is operationalized as a function of *(request) response time*. This response time is defined as the time period that elapses between request arrival and start of service (see Larsen et al., 2007). Clearly, resulting response time correlates with customer inconvenience. In marketing literature, two main approaches for modeling customer inconvenience depending on elapsed response times can be found. Davis and Maggard (1990) propose a linear increase in customer inconvenience with resulting response time. In contrast, Kristensen et al. (1992) define a quadratic dependency.

A typical real-world application that motivates this work is the subsequent delivery of newspapers. This process occurs when a subscriber has not received his newspaper due to a delivery failure or thievery. It is of particular importance to compensate the subscriber either by voucher or subsequent delivery. According to Tax et al. (1998), customer satisfaction relates to how a company deals with complaints. Smith et al. (1999) state that in case of a delivery problem, a company has the possibility to either restore customer satisfaction or risk losing the customer. Fast subsequent delivery allows market differentiation for a publishing house. This is of particular importance in times of diminishing revenues from advertisements due to an emerging market share of Internet

* Corresponding author. Tel.: +49 202 439 2442; fax: +49 202 439 3434.
E-mail addresses: fferrucci@winfor.de (F. Ferrucci), sbock@winfor.de (S. Bock), michel.gendreau@cirrelt.ca (M. Gendreau).
URLs: http://www.winfor.de (F. Ferrucci), http://www.winfor.de (S. Bock), http://www.cirrelt.ca (M. Gendreau).

0377-2217/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ejor.2012.09.016

advertisements. Bieding et al. (2009) underscore the importance of applying on-line algorithms to efficiently manage the subsequent delivery of newspapers. Also repairmen companies face strong competition and are bound to service level agreements (SLAs) with customers (see van de Klundert and Wormer, 2010). Customers will call the company when a problem that is covered by the SLA occurs. If a repairman company does not fulfill a request within the time period defined in the SLA, high penalty costs occur. Moreover, servicing customers before the end of the SLA time period increases customer loyalty and there may be additional monetary motivations which result when both parties share saved costs. In contrast to the subsequent delivery of newspapers, the time periods required at customer locations are neither homogeneous nor predetermined. Another related application area is the taxi business where people frequently demand instant service so that quality of service is also measured by response times. Differences from the previously described applications are in the transportation process itself which usually has to be modeled as a dial-a-ride-problem (DARP, see Cordeau and Laporte, 2007) where a request consists of two locations and the service time needed for transporting a customer contains uncertainty.

In literature, many deterministic real-time control approaches for the DVRP that only react to already arrived requests can be found (Savelsbergh and Sol, 1998; Gendreau et al., 1999; Ichoua et al., 2000; Ghiani et al., 2003; Giaglis et al., 2004). In contrast to these deterministic approaches, recent research has integrated stochastic elements into DVRPs. It has been shown that the exploitation of knowledge about past requests enables considerable improvements (Bent and van Hentenryck, 2004b; Ichoua et al., 2006; Hvattum et al., 2006, 2007). Bent and van Hentenryck (2004b) introduce a sampling-based approach that is extended in Bent and van Hentenryck (2004c,a, 2005, 2007). Further information is provided in van Hentenryck and Bent (2009). Ichoua et al. (2006) extend the approach of Gendreau et al. (1999) by exploiting knowledge about future request arrivals. A vehicle waits at the location of its last request if the probability of new, nearby requests is high. Hvattum et al. (2006) propose a sample scenario hedging heuristic that uses statistical information about future requests. In Hvattum et al. (2007), the authors improve this heuristic by integrating it into a branch-and-regret heuristic. In order to classify the quality of available stochastic information, Psaraftis (1995) and Larsen et al. (2007) propose the distinction between known requests (known-deterministic), given uncertain stochastic information (based on forecasts), given probabilistic stochastic knowledge (a prescribed distribution is known), or the case of no stochastic knowledge about future requests. Furthermore, many real-time control approaches for the DVRP examine the objective to minimize the number of unserviced customers within fixed time windows. Note that this objective does not focus on service level quality by minimizing inconvenience of customers as considered for repairman service processes by Westphal and Krumke (2008) and van de Klundert and Wormer (2010). While Westphal and Krumke (2008) propose a quadratic objective function, customer inconvenience linearly increases with the response time in van de Klundert and Wormer (2010).

In order to attain a significant practical applicability, the real-time control approach proposed in this paper combines the following contributions:

- *Pro-active real-time control approach*. Stochastic knowledge is used in a new pro-active real-time control approach by integrating dummy customers. The approach coordinates the utilization of the vehicles in accordance with expected future requests. It actively guides them into request-likely areas before requests arrive there. Note that this approach not only determines the waiting positions for idle vehicles but also pro-actively coordinates busy vehicles. In order to handle the high degree of dynamism (cf. Larsen et al., 2002; Ichoua et al., 2007) of the considered problem, a quick generation of efficient tour plans is mandatory. Therefore, a Tabu Search with a variable neighborhood structure is proposed. Depending on previously explored solutions, diversifying and intensifying operators are applied. Furthermore, vehicles move on a fully-detailed urban road network in order to increase the degree of modeled realism.
- *Universally applicable integration of stochastic knowledge*. In contrast to many other approaches, stochastic knowledge about future request arrivals is solely generated from past request information without assuming existing distributions. In order to ensure an efficient applicability of the stochastic knowledge, specific quality criteria are developed.
- *Classification of data quality*. Based on empirical studies with various test settings, criteria for measuring request data quality are derived that allow the identification of scenarios in which the proposed approach (depending on the available fleet size) can be efficiently applied. We introduce the degree of structural diversity (*dosd*) as a general measure of existing variability in request data sets and prove its significance.
- *Integration of time urgency measures into the objective function*. Since requests have to be serviced as soon as possible, many existing vehicle routing approaches with time windows frequently do not map the existing urgency adequately. Therefore, the proposed approach pursues the minimization of customer inconvenience. The effects of using a linear as well as a quadratic dependency between response time and customer inconvenience are investigated.

The remainder of this paper is organized as follows. Section 2 illustrates the considered problem. Section 3 introduces the deterministic real-time control approach. Section 4 explains how stochastic knowledge is generated and integrated into the real-time control approach. Section 5 presents the Tabu Search method used for solving static problems. In Section 6 we analyze the computational results. Based on these results, Section 7 introduces the degree of structural diversity and shows its statistical significance. We conclude in Section 8 by summarizing our main results and discussing further research fields.

2. Problem description

In the considered problem a daily distribution process is controlled in real-time. This process is defined as a Dynamic Vehicle Routing Problem with Soft Time Window constraints (DVRPSTW) comparable to the one considered by Gendreau et al. (1999): A set of homogeneous vehicles $K = \{1,\dots,m\}$ located at one depot delivers goods to a set of customer requests $R = \{1,\dots,n\}$ that occur over the day. The set of known requests at time point τ is defined by $R_\tau = \{i \in R | a_i \leqslant \tau\}$. Each request $i \in R$ has an arrival time a_i and can be directly serviced upon arrival so that its time window e_i opens at a_i. The length of the time window is defined by a maximum allowed response time R^{mrt} that is the same for all requests. If service begins after this time period, high penalty costs occur. The use of this maximum allowed response time was motivated by a project with a publishing house which performs the subsequent delivery of newspapers. In this project, it was revealed that the subsequent delivery should be performed as quickly as possible but no customer request should wait longer than a defined time period. After evaluating different approaches, the utilization of R^{mrt} in combination with high penalty costs turned out to be very effective. In doing so, customer requests which are located outside the city center are also serviced within the given maximum response time limits. Without using high penalty costs, it was observed that

those customer requests were postponed for a long time in favor of an earlier service of several inner city requests.

The service time s_i of request i is the same for all requests and is set to $R^{st} = 60$ seconds. The point in time when service begins at request i is defined by y_i. The problem at τ consists of determining the tour plan that assigns the set of currently n_τ unserviced requests $R_\tau^U = \{i \in R_\tau | y_i > \tau\}$ to the vehicles with the minimum objective function value. Note that besides known requests, R_τ^U comprises dummy customer requests that have to be integrated into the tour plan in order to guide vehicles into request-likely areas. Since dummy customer requests represent expected future requests, their time window always starts in the future. Each request i obtains an individual factor w_i by which it is weighted in the objective function. While this weight is 1 for real requests, it is set to the occurrence probability of requests in the corresponding request-likely area. Furthermore, capacity constraints can be regarded as negligible, since a sufficient amount of goods can be loaded onto the vehicles at the depot.

2.1. Objective function

The objective function aims at minimizing the total customer inconveniences that are operationalized by a function of request response times. Exceeding the maximum allowed response time $R^{mrt} = 3600$ seconds results in a prohibitive increase of customer inconvenience (R^{pen}). The objective function is defined by

$$\min z = \sum_{i \in R_\tau^U} \left(w_i \cdot \left(\underbrace{F(t_i)}_{\text{Variable inconvenience}} + \underbrace{R^{pen} \cdot \Theta(t_i - R^{mrt})}_{\text{Lateness inconvenience}} \right) \right),$$

with $t_i = y_i - e_i$, $i \in R$ and $\Theta(x) = 1$ if $x > 0$ and 0 otherwise. F is defined either as a linear or a quadratic customer inconvenience function (see Fig. 1).

Linear customer inconvenience (linear2X). In this case, F is defined as $F(t_i) = \frac{1}{R^{mrt}} \cdot (\min(t_i, R^{mrt}) + 2 \cdot \max(0, t_i - R^{mrt}))$. Note that we double the steepness after the maximum allowed response time R^{mrt} in order to still consider those requests with a high priority.

Quadratic customer inconvenience (quadratic). F is defined as $F(t_i) = \left(\frac{t_i}{R^{mrt}}\right)^2$. Since the steepness increases implicitly, no adaptation is required.

In both cases a response time equal to R^{mrt} results in a customer inconvenience contribution of 1.0 whereas exceeding R^{mrt} is penalized by $R^{pen} = 100$.

2.2. Road network and digraph

In order to accurately simulate vehicle movements on roads, a real urban road network $\widetilde{R} = (\widetilde{N}, \widetilde{A})$ is used with $\widetilde{N}$ and $\widetilde{A}$ denoting its nodes and arcs, respectively. In order to model the current situation in the transportation network at time τ, a digraph that maps all locations that are relevant at τ is used. It comprises all request locations including the depot as well as all current vehicle positions. Let $G = (N, A)$ be a digraph where $N = \{0, \ldots, n_\tau + m\}$ is the associated set of nodes and $A = (N \times N)$ the set of arcs. Node 0 corresponds to the depot node $\tilde{n}_{depot}$ and nodes $j = 1, \ldots, n_\tau$ correspond to the set of unserviced request locations R_τ^U at time τ. It holds that $N \subset \widetilde{N}$, i.e., each request is located at a crossing or at a road endpoint in the road network $\widetilde{R}$. For each vehicle $k \in K$, node $(n_\tau + k) \in \widetilde{N}$ represents the location of vehicle k at τ in the road network. Note that if vehicle k is not at a node of $\widetilde{R}$ at τ but on an arc $\tilde{a} \in \widetilde{A}$, $(n_\tau + k) \in \widetilde{N}$ represents the node of the road network that vehicle k will reach next. Furthermore, $v_{k\tau}^{\tilde{s}t}$ represents the *availability time* of vehicle k. It is larger than 0 if the vehicle is busy with traveling or with servicing a request at τ and defines when it is available at node $(n_\tau + k) \in \widetilde{N}$. A non-negative value t_{ij} is associated with each arc $(i,j) \in A$. It represents the shortest time needed for a vehicle to travel from node i to node j.

2.3. Allowance of vehicle en-route diversion

In our problem setting, the first request currently assigned to a tour of a vehicle is allowed to be changed if this leads to a better solution (see also Ichoua et al., 2000). This enables a more direct reaction to newly incoming requests. In our scenario this is likely to be beneficial since new arriving requests may be serviced very quickly if a vehicle only needs to perform a small detour on its way to its currently assigned first request.

2.4. Presence of historical request information

As mentioned above, information about past request arrivals can be used in order to anticipate future requests of the next day. Hence, in our problem setting, information about future requests is only available with uncertainty and does not follow prescribed probability distributions that are known to the system. This is in line with the classification of Psaraftis (1995) and Larsen et al. (2007) and is representative for many practical applications. By using past request information, vehicles are distributed over the considered service area in order to be prepared for expected future arriving requests. Consequently, future requests can be serviced earlier leading to an improved solution quality.

3. The deterministic real-time control approach

Since the considered routing problem is strongly NP-hard, sophisticated plan adaptations required to integrate newly arriving requests cannot be carried out in zero time. In order to allow a plan

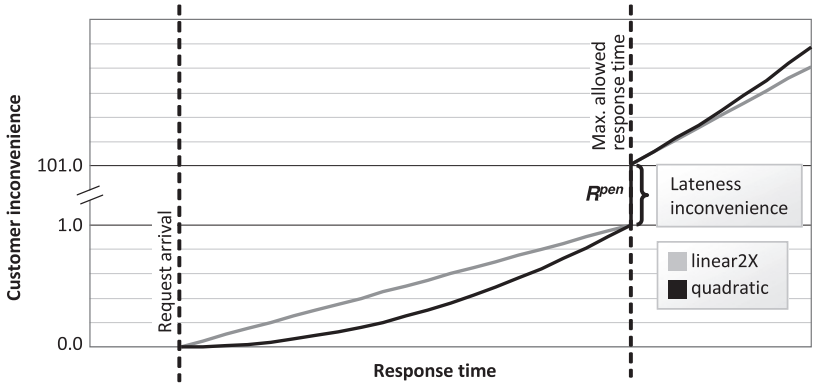

Fig. 1. Different evolutions of linear2X and quadratic customer inconvenience.

adaptation simultaneous to the execution of the transportation process, we apply a concept related to the one proposed by Bock (2010). In this concept, real-time control is conducted by solving consecutive static problem instances. The instances arise from the ongoing transportation process and are generated at intervals of t^a time units each, denoted as *anticipation horizons*. The *relevant plan* currently in execution is P^r_τ, where τ denotes the time at the beginning of the considered anticipation horizon. Furthermore, P^t_τ defines a *theoretical plan* that is optimized during the anticipation horizon and P^{bt}_τ denotes the best plan found during the optimization.

At the beginning of each anticipation horizon, all requests that were buffered during the previous anticipation horizon are integrated thus generating P^r_τ. In order to handle the simultaneity of real-time plan adaptation and process execution, the static problem at τ is derived from a snapshot of the system state at the end of the current anticipation horizon by pre-simulating P^r_τ for t^a time units. Consequently, P^t_τ is directly implementable at the end of the anticipation horizon. After the pre-simulation, the remaining time in the anticipation horizon is used for optimizing P^t_τ. Clearly, since P^t_τ contains only decisions that are taken after the elapse of the current anticipation horizon, all decisions taken by P^r_τ are executed as planned for the current anticipation horizon. Hence, the assignment of a request to a vehicle becomes fixed if service at this request starts in P^r_τ before $\tau^+ = \tau + t^a$. Plans are updated and replaced using a method similar to the one proposed by Bock (2010).

In the approach, a *vehicle waiting strategy* is used. Here, an idle vehicle waits at its current location for new requests. Computational experiments show that this strategy is superior compared to an immediate return to the depot.

4. Integrating stochastic knowledge into the real-time control approach

In this section, methods for generating stochastic knowledge about expected future requests are proposed. The stochastic knowledge is integrated into the real-time control approach in order to guide vehicles to request-likely areas instead of passively waiting for new requests at the last customer location. This enables an earlier service of future requests in those areas and is advantageous with regard to the considered objective function. We designate this extended approach as the *pro-active real-time control approach*.

4.1. Turning stochastic knowledge into dummy customers

Stochastic knowledge is generated by analyzing past request information of the last n^f days that are assumed to be representative for the next day. Subsequently, this knowledge is integrated into the deterministic real-time control approach by the generation of *dummy customer requests* (or short dummy customers). Dummy customer-related approaches have been used among others by, e.g., Ichoua et al. (2006) and Bent and van Hentenryck (2007). In contrast to many of these approaches, our approach generates dummy customers by the application of a sophisticated offline procedure before the execution of the transportation process. This procedure provides stochastic knowledge of customizable quality about expected future requests. Since this stochastic knowledge defines the temporal and spatial position of the generated dummy customers, only one scenario comprising these high-quality dummy customers has to be solved during the time-critical adaptations of the transportation process. Note that this is a significant advantage compared to many previous approaches which require solving numerous sample scenarios simultaneous to the execution of the transportation process. Consequently, in order to provide useful decision support, these approaches have to generate stochastic knowledge in real-time under high time pressure which may limit their practical applicability and efficiency.

Dummy customers are generated in an offline step. Therefore, they are directly integrated into the initial tour plan. Dummy customers and real requests are handled similarly except in the following three aspects:

- In contrast to real requests, whose time window directly begins with their arrival, time windows of dummy customers open at a specified time point derived from past request information.
- Dummy customers guide vehicles to request-likely areas and keep them there for a specified amount of time. This amount of time is determined by the total sum of service and travel time required to fulfill expected requests in a corresponding temporal and spatial area. Hence, the service time is set accordingly.
- A dummy customer i represents requests expected with some uncertainty. Hence, it is considered in the objective function with $w_i < 1$. Thus, the pro-active real-time control approach prioritizes real requests that have to be serviced with certainty.

4.2. Segment-based clustering and placement of dummy customers

In order to derive reliable stochastic knowledge about expected future requests, the service area is divided into *segments*. According to Fig. 2, each segment $s \in S$ has a quadratic spatial (DC^{se}) and a temporal (DC^{te}) extension. For each $s \in S$, request arrivals are modeled by a time–space Poisson process p_s. These processes may vary from segment to segment but are constant from day to day. Request arrivals are assumed to be independent of each other since these processes are memoryless. The rate parameter $\lambda(p_s)$ is calculated by the average number of past request arrivals in s over the last n^f days.

Dividing the service area into small segments often leads to low rate values $\lambda(p_s)$. In order to allow reliable forecasts, adjacent segments are combined into *clusters*. Specifically, n time–space Poisson processes $p_1, \ldots, p_n$ can be combined into a new compound time–space Poisson process $p^* = \bigcup_{i=1}^{n} p_i$, with $\lambda(p^*) = \sum_{i=1}^{n} \lambda(p_i)$ until p^* obtains a rate parameter that is sufficiently high, thereby justifying the assumption that requests will arrive in the cluster. Segments are combined into clusters so that the following quality criteria are fulfilled:

(i) A cluster does not exceed a predefined maximum spatial and temporal extension DC^{mse} and DC^{mte}, respectively.
(ii) The sum of the rate values of assigned segments is at least $DC^{min\lambda}$.
(iii) Certain further quality aspects are met (this will be described later on).

Preliminary tests revealed that clusters of desired quality cannot be generated by several well-known approaches such as the k-Means algorithm (cf. MacQueen, 1967). First, k-Means requires the number of clusters to be specified. In our case, this is inappropriate since we aim at maximizing the number of generated clusters which depends on the quality of the past request information. Moreover, considering criteria (i), cluster extensions cannot be controlled and clusters generated by k-Means are rarely compact and often irregular (see Fig. 3). Hence, we present a new cluster generation approach.

Our approach works in two phases. In the first phase all valid clusters which meet the above criteria are generated by considering all *bases*. A base comprises either 1, 2, or 4 segments adjacently located at the same temporal level (see Fig. 3) and together form a convex cluster c_{temp} with a temporal height of one segment. This

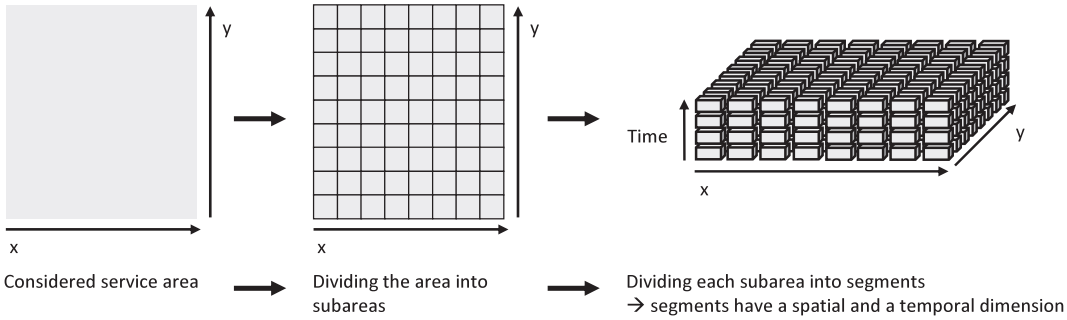

Fig. 2. Dividing the service area into subareas and segments.

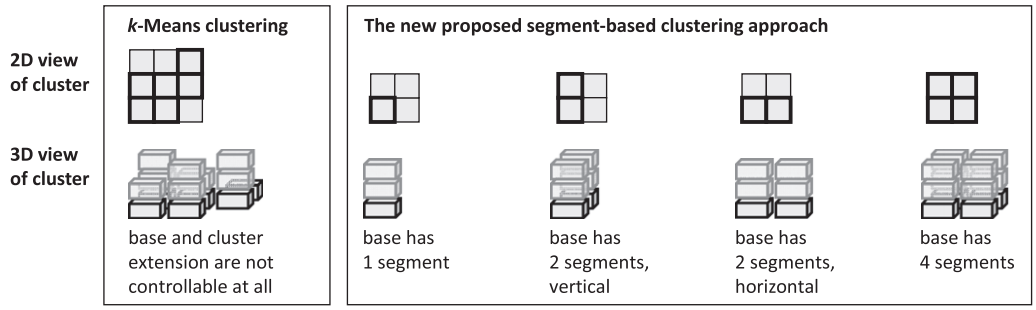

Fig. 3. The *k*-Means clustering algorithm and the new cluster generation approach with different convex bases.

provides a spatial cluster shape that fulfills the first part of criteria (i). In order to fulfill criteria (ii), segments located at the successive temporal level are iteratively added to c_{temp} (see Fig. 4) until the calculated rate parameter $\lambda(c_{temp})$ is at least $DC^{min\lambda}$ or the temporal height of c_{temp} exceeds DC^{mte}. In the latter case, c_{temp} is discarded, since it violates the second part of criteria (i). Otherwise, further quality aspects previously mentioned in criteria (iii) are checked by conducting two further steps.

First, the spatial barycenter of past requests observed within c_{temp} over the last n^f working days is mapped on the road network by determining the closest node n'. Since n' may be positioned within an area consisting of roads with a low speed limit, it is relocated to node $n_{c_{temp}}$ that fulfills two requirements: (a) it is located within a maximum travel time radius $DC^{radiusTT}$ of n' and (b) it is connected with a road of a minimum desired speed limit in order to provide a reasonable road connection. Node $n_{c_{temp}}$ is determined by conducting a Dijkstra search that starts at n' and is limited by $DC^{radiusTT}$. If such a node does not exist in this radius, the closest node with largest speed limit is chosen. This node is the location of the dummy customer of c_{temp}. If the average travel time c_{temp}^{avgTT} from $n_{c_{temp}}$ to all past requests observed within c_{temp} over the last n^f working days exceeds a maximum average travel time $DC^{maxAvgTT}$, c_{temp} is discarded since the expected detour in c_{temp} is too large to enable a precise forecast. In order to verify whether request arrivals can be adequately modeled by Poisson processes, the Poisson quality of c_{temp} is analyzed in a second step. Specifically, it is checked whether there exist significant deviations between the number of actual customer request arrivals in c_{temp} over the considered n^f working days and those that are expected according to the Poisson distribution with rate parameter $\lambda(c_{temp})$. This is performed by Pearson's Chi-Square Goodness-of-Fit test (cf. Kvam and Vidakovic, 2007, p. 155 et seq.). Results indicate that a type I error of $\alpha = 0.40$ generates clusters with a sufficient Goodness-of-Fit quality. Specifically, it was shown that the implemented method is capable of efficiently identifying request data sets which do not fulfill the aforementioned Poisson quality requirement. Note that the additional postulate of Poisson processes that request arrivals occur independent of each other (cf. Ross, 2010, p. 313) can be analogously checked. However, due to the small temporal extension of a cluster, we refrain from performing this test. This is because the required accuracy in request arrival times is frequently not fulfilled by practical request data sets and therefore would lead to an unfavorable rejection of clusters.

If c_{temp} fulfills all quality criteria, it is added to the set of valid clusters C. Let $PS_\lambda(x)$ denote the PMF of the Poisson distribution. The dummy customer i belonging to c_{temp} gets the weight $w_i = P(X \geqslant 1) = 1 - PS_{\lambda(c_{temp})}(0)$. Its service time is defined by $s_i = \left(R^{st} + c_{temp}^{avgTT}\right) \cdot \lambda(c_{temp})$. Since this is the expected amount of time that a vehicle will spend in c_{temp}, dummy customers increase the robustness of generated tour plans. Thanks to this increased robustness, substantial tour adaptations under high time pressure become less likely so that sophisticated tour plans can be more frequently implemented as planned. In combination with actively guiding vehicles to request-likely areas in order to service future expected requests earlier, this is likely to lead to better results.

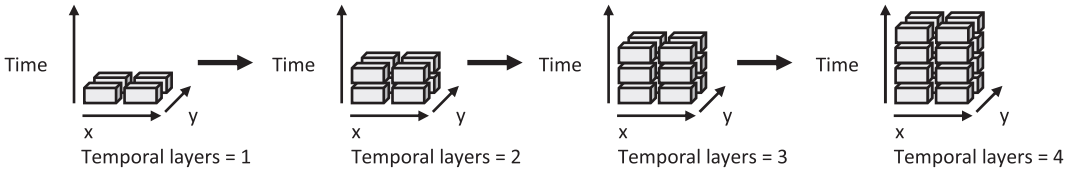

Fig. 4. Steps of the cluster generation scheme generating one valid cluster using a base consisting of four segments.

After generating all valid clusters, the maximum number of non-overlapping clusters is selected by solving the following MIP. The set $S_C \subseteq S$ comprises all segments which are assigned to clusters of the set C. The binary parameter c_{is} is set to 1 if cluster $i \in C$ contains segment $s \in S_C$. Moreover, c_i^{start} denotes the temporal level at which cluster i starts. The binary decision variable x_i is 1 if cluster $i \in C$ is selected, 0 otherwise. Parameter M is a big number.

$$\max z = \sum_{i \in C} (M \cdot x_i - c_i^{start} \cdot x_i) \quad (1)$$

$$\text{s.t. } \forall s \in S_C : \sum_{i \in C} c_{is} \cdot x_i \leqslant 1 \quad (2)$$

$$\sum_{i \in C} x_i \leqslant C^{max} \quad (3)$$

The primary goal of the objective function (1) is the maximization of the number of selected clusters. The secondary goal is to start clusters at the earliest point in time in order to support an earlier service of expected requests. Constraints (2) ensure that all segments are assigned to at most one selected cluster. Constraint (3) imposes a theoretical upper bound on the number of selectable clusters. This restriction was defined since it allows the MIP solver to generate tighter bounds. For the computational experiments, this bound is set to $C^{max} = \left\lfloor \frac{\sum_{s \in S_C} \lambda(p_s)}{DC^{min\lambda}} \right\rfloor$. After solving the MIP, a corresponding dummy customer is added to the set of initial requests for each selected cluster.

4.3. Dynamic updating of dummy customers during tour plan processing

From the moment when the current time τ reaches the beginning of cluster c_i, the parameter values of dummy customer i change since the remaining rate parameter decreases with τ. Hence, for a new static problem generated at τ, the remaining rate value is updated to $\lambda(c_i, \tau^+)$. Clearly, for each cluster with $c_i^{start} \geqslant \tau^+$, it holds that $\lambda(c_i, \tau^+) = \lambda(c_i)$. Moreover, for cluster c_i the start time of its remaining part at τ^+ is defined as $c_i^{start}(\tau^+) = \max(c_i^{start}, \tau^+)$. Based on τ^+, parameters of dummy customer i are updated as follows:

- Beginning of time window (e_i): Since dummy customer i represents expected future requests, e_i is set to the expected arrival time of the next request in cluster c_i assuming that at least one more request will occur in c_i. With c_i^{end} denoting the end time of the cluster c_i, e_i is updated as follows:

$$e_i(\tau^+) = c_i^{start}(\tau^+) + \left(c_i^{end} - c_i^{start}(\tau^+)\right) \cdot \frac{\sum_{n=1}^{\infty} PS_{\lambda(c_i,\tau^+)}(n) \cdot \frac{1}{n+1}}{1 - PS_{\lambda(c_i,\tau^+)}(0)}$$

The expected arrival time of the next request can be derived by determining the average expected time position of the next request over all scenarios with $1, 2, 3, \ldots, \infty$ requests that arrive in c_i in total. Thus, due to the impact of $e_i(\tau^+)$ on the objective function value, it is likely that in generated tour plans dummy customers are not visited before $e_i(\tau^+)$.

- Service time (s_i): Since the service time of dummy customer i is the sum of average travel and service time for fulfilling expected requests in cluster c_i, this parameter is updated by $s_i(\tau^+) = \left(R^{st} + c_i^{avgTT}\right) \cdot \lambda(c_i, \tau^+)$.
- Weight factor (w_i): For dummy customer i, w_i is updated according to the probability at τ^+ that at least one more request will arrive in cluster c_i. Therefore, it holds that $w_i(\tau^+) = P(X \geqslant 1) = 1 - PS_{\lambda(c_i,\tau^+)}(0)$.

A dummy customer is removed from $R^U_{\tau^+}$ when it becomes unlikely that further requests will arrive in c_i. This time point is reached when $\lambda(c_i, \tau^+)$ becomes smaller than $DC^{\lambda rem}$. Note that this time point can be calculated in advance.

4.4. Extended vehicle waiting strategy with regard to dummy customers

During the execution of the transportation process, scenarios may occur where a dummy customer i is scheduled as the next request but due to a much later opening of e_i, its service is planned in the far future. Using the vehicle waiting strategy defined in Section 3, the vehicle would immediately drive to dummy customer i. This can lead to unwanted scenarios where, for instance, vehicles are guided to remotely located areas too early and therefore are not available for servicing other requests in more central regions. Hence, an extended version of the waiting strategy is used. If a vehicle's next request is a dummy customer i, the vehicle waits at its current position as long as the remaining time to c_i^{start} is greater than the travel time required to reach dummy customer i.

5. The Tabu Search solution method

In each anticipation horizon starting at τ, a Tabu Search metaheuristic (see Glover, 1989; Glover and Laguna, 1998) is applied to the corresponding problem instance for $t^a - t_\tau^{calc}$ time units. In order to integrate intensification and diversification aspects, the Tabu Search procedure switches between different stages that control the applied neighborhood operators. If a defined number of iterations has been performed without finding a new best solution, the Tabu Search algorithm switches into the next higher stage and continues its search with a different neighborhood operator. Neighborhood operators on lower stages intensify the search in vicinity of the current solution, whereas higher stages apply more diversifying operators. If a new best solution is found, the search returns to the first stage. The following neighborhood operators are applied. Used random numbers are generated by the procedure of Park and Miller (1988):

Within Tour Insertion (WTI). WTI removes a single request and reinserts it at its best position on the same tour if this leads to a solution improvement. For each tour the best relocation that is not tabu is implemented.

Relocate (REL). REL and WTI work similarly but REL evaluates all positions that are on tours other than the one a request is currently assigned to.

MultiRelocate (MREL). MREL removes n requests where n is randomly drawn out of the interval $[MREL^{reqMin}, MREL^{reqMax}]$. They are successively reinserted at their least cost positions by considering all possible reinsertion permutations. The n requests are randomly chosen out of the $n \cdot MREL^{reqSelRatio}$ requests with the highest objective function value contribution. In MREL, the entire procedure is repeated c times, (c randomly chosen out of the interval $[MREL^{execMin}, MREL^{execMax}]$). The best non-tabu move is executed.

Large Neighborhood Search (LNS). LNS reinserts n requests in the sequence of non-increasing cost contributions where n is set to $\max\left(2, m \cdot \left|R^U_{\tau^+}\right|\right)$ and m is randomly drawn out of the interval $[0, LNS^{reqRemFactor}]$. The n requests are randomly chosen out of the $\min\left(\left|R^U_{\tau^+}\right|, n \cdot LNS^{reqSelRatio}\right)$ requests with the highest objective function value contribution. This is repeated c times where c is randomly chosen out of the interval $[LNS^{execMin}, LNS^{execMax}]$.

Exchange Between Tours (XBT). XBT exchanges two requests of two tours. After evaluating all combinations, the best non-tabu move is performed.

A fingerprint of the tour plan is applied as the *tabu-active attribute*. It is implemented by combining two quickly computable checksums CRC-32 (cf. Moon, 2005, p. 147 et seq.) and Adler-32 (cf. Deutsch and Gailly, 1996). Two tour plans are identified as identical if both checksums are equal. Since tests indicate that this

Table 1
Stage-based selection scheme.

Stage	Operator	Switch after number of iterations
1	WTI	0 (only accept improvements)
2	REL	10
3	MREL	10
4	LNS	1000
5	XBT	5

Table 2
Applied parameter values for MREL and LNS.

Parameter	Value	Parameter	Value
$MREL^{reqMin}$	2	$LNS^{reqRemFactor}$	0.75
$MREL^{reqMax}$	3	$LNS^{reqSelRatio}$	1.5
$MREL^{reqSelRatio}$	3	$LNS^{execMin}$	1
$MREL^{execMin}$	10	$LNS^{execMax}$	2
$MREL^{execMax}$	20		

allows a reliable identification of tour plans, a tabu state is never revoked.

The initial tour plan is constructed at the beginning of the day by an iterative least cost insertion of all real requests known at this point in time followed by the dummy customers. Dummy customer requests are integrated in sequence of non-decreasing time window start times.

In each iteration of the Tabu Search procedure, the applied neighborhood operator is chosen by using the deterministic selection scheme depicted in Table 1. The applied parameter values of MREL and LNS are provided in Table 2.

6. Computational experiments

In this section, the performance of the proposed deterministic and pro-active real-time control approach is analyzed. The efficiency of the Tabu Search approach which is iteratively applied in both real-time control approaches is evaluated in a preliminary study on individual static problem instances. A comparison with an exact solution approach reveals that the Tabu Search approach attains near-optimal or optimal solutions for all instances. Using a fleet of 10 vehicles, the average solution quality of the Tabu Search procedure is only 0.017% below the lower bound values computed by the approach of Westphal and Krumke (2008). Note that the computational time that is available for the Tabu Search procedure is restricted to 10 seconds. Hence, applying this approach to static instances during the real-time process allows a sufficiently fast and efficient plan adaptation.

6.1. Test environment, test parameters, and the evaluated request data set

All experiments are executed on the road network of Dortmund, a German medium-sized city, by using a developed discrete-event based simulator and fleet sizes of 8, 10, and 12 vehicles. The considered region has a size of 22.5 km $\times$ 20 km = 450 km^2. Before the beginning of the transportation process, the initial tour plan is optimized by the Tabu Search approach for 120 seconds. All real-time control approaches were implemented in Delphi and the experiments were run on Pentium D 2.8 GHz CPU (2.5 GB RAM) computers. Suitable parameter values for dummy customer generation and integration are empirically derived in preliminary tests. The number of past days used is set to $n^f = 60$ and the extension of each segment is set to $DC^{se} \times DC^{te}$ = (2.5 km $\times$ 2.5 km) $\times$ 1 minute. The maximum spatial cluster extension DC^{mse} is set to 2 by 2 segments while the maximum temporal cluster height (DC^{mte}) is 15 minutes (15 segments). The maximum travel time radius $DC^{radiusTT}$ is set to 300 seconds whereas the maximum average travel time within a cluster $DC^{maxAvgTT}$ is set to 650 seconds. For $DC^{min\lambda}$, five values are evaluated: $DC^{min\lambda} \in \{1.0, 1.2, 1.5, 1.8, 2.0\}$ so that the minimum occurrence probability of at least one request in a cluster amounts to (63.21%, 69.88%, 77.69%, 83.47%, 86.47%). Two values are evaluated for $DC^{\lambda rem}$:$DC^{\lambda rem} \in \{0.50, 0.25\}$. Hence, a dummy customer is removed when the arrival probability of at least one more request becomes lower than 39.45% or 22.12%, respectively.

The computational experiments are conducted on request data sets that are generated according to characteristics observed in a case study on the subsequent delivery process of a German newspaper publishing company. By analyzing request information, we identified that on average 150 requests arrive per day. Vehicles leave the depot at 7 am and the majority of the subsequent delivery requests arrive before 11 am. Therefore, in each daily instance these four hours are evaluated. Since more than 90% of all requests arrive after 7 am, a high degree of dynamism is present (see Larsen et al., 2002).

In the considered request data sets, arrival times and locations of incoming requests are generated by time–space Poisson processes. For this purpose, the service area is divided into a set of P disjunctive quadratic regions of equal size $a_1, \ldots, a_P$ while the time period for new arriving requests from 6:45 am to 11 am is divided into Q different time slices $t_1, \ldots, t_Q$ of individual length each. Parameter $\lambda(a_i, t_j)$ determines the request arrival rate of a time–space Poisson process in region a_i during time slice t_j. Given a time interval t_j, the arrival time of the next request is determined using a Poisson process with rate parameter $\lambda(t_j) = \sum_{i=1}^{P} \lambda(a_i, t_j)$. The region in which this new request will occur is determined according to the region probability values $p(a_i, t_j) = \lambda(a_i, t_j)/\lambda(t_j)$. Within the chosen region, the location of the new request is uniformly drawn according to the properties of time–space Poisson processes.

Preliminary experiments indicate that the rate parameters $\lambda(a_i, t_j)$ significantly influence the quality of the stochastic knowledge that can be generated from past request information. Hence, resulting variations of request arrival rates are identified as a crucial factor for an efficient application of derived stochastic knowledge. In what follows, this factor is denoted as *structural diversity*. We distinguish between two dimensions of structural diversity: While RegionDiversity (*RD*) measures the regional variance of request arrivals in the service area within a considered time slice, TimeDiversity (*TD*) gives the variance of request arrivals along the time elapse in a considered region of the service area. A suitable testbed for a detailed analysis of different levels of spatial and temporal structural diversity is provided by systematically designing different settings. All settings are iteratively generated by starting from an initial setting with a maximum level of structural diversity (*RD* = *TD* = 1.00). The structural diversity of this initial setting results from individual intensity rate parameters $\lambda(a_i, t_j)$ and is iteratively reduced in subsequently generated settings by applying a linear averaging that balances the rate parameters $\lambda(a_i, t_j)$ accordingly. By using a step-size of 0.25 and completing with a final setting *RD* = *TD* = 0.00, we obtain 25 settings altogether. Apart from negligible distortion effects caused by the real-world character of the road network, setting *RD* = *TD* = 0.00 produces a complete uniform distribution of request arrivals during the considered planning horizon. In all settings, the number of regions is set to $P = 18$ and the number of time slices to $Q = 5$. In order to generate request data with a degree of dynamism comparable to the considered real-world application, over 90% of all requests occur during the execution of the transportation process. Each tested request data set consists of 30 instances.

6.2. Performance evaluation of the deterministic real-time control approach

In this section, the deterministic real-time control approach that is introduced in Section 3 is evaluated with t^a set to 20 seconds and is designated as ROLLING20 in what follows. It is tested in direct comparison with two other approaches. The approach ZEROTIME is a pure zero-time adaptation that integrates newly incoming requests into the tour plan using a least cost insertion heuristic. The second approach TIMELIMIT10 extends ZEROTIME by additionally applying the Tabu Search procedure for $\frac{t^a}{2} = 10$ seconds when new requests have arrived during the previous anticipation horizon.

The results in Table 3 show that the sophisticated real-time control approaches TIMELIMIT10 and ROLLING20 significantly outperform ZEROTIME for both tested objective functions. Most improvements are achieved in scenarios with strongly limited resources (8 vehicles) since a nearly optimal usage of the limited capacities (provided by TIMELIMIT10 and ROLLING20) in these scenarios is gaining more importance. In accordance with the results achieved by Gendreau et al. (1999) and Bock (2010), it can be concluded that sophisticated real-time control approaches are required for an efficient execution of complex transportation processes. By directly comparing the results of TIMELIMIT10 and ROLLING20 with each other (see Table 3), it can be observed that ROLLING20 does not produce any significant improvements for the considered application. This underlines that the Tabu Search algorithm is able to quickly attain nearly optimal solutions as stated at the beginning of Section 6. Since ROLLING20 does not use the increased computational times efficiently, in all following experiments the solution method is applied as in TIMELIMIT10.

6.3. Performance evaluation of the pro-active real-time control approach

This subsection directly compares the results of the pro-active real-time control approach (PROACTIVE10) with the results generated by the deterministic approach TIMELIMIT10 for all structural diversity settings in order to analyze under which circumstances the integration of stochastic knowledge is useful. Regarding the two tested values of the parameter $DC^{\lambda rem}$, preliminary tests indicate that an earlier removal of dummy customers is more promising. Specifically, setting $DC^{\lambda rem}$ to 0.50 allows a more flexible tour plan adaptation. Specifically, using a smaller value for $DC^{\lambda rem}$ leads to vehicles being kept too long in areas where the probability of a future request arrival has become too low. Hence, in the following experiments, $DC^{\lambda rem}$ is set to 0.50.

The results in Table 4 illustrate the improvements attained by PROACTIVE10 in comparison with TIMELIMIT10 for all structural diversity settings. Moreover, the number of worse test instances out of the 30 evaluated ones is illustrated. According to the objective function described in Section 2.1, solutions generated by PROACTIVE10 and TIMELIMIT10 are compared according to the number of late requests (primary objective) and the total customer inconvenience (secondary objective). Therefore, a solution generated by PROACTIVE10 is denoted as worse if either (a) it comprises more late requests or (b) it comprises an equal number of late requests but a larger total variable customer inconvenience than the one generated by TIMELIMIT10. The results show that the level of structural diversity mainly triggers the solution quality of PROACTIVE10. Specifically, the more structural diversity is present, the more PROACTIVE10 benefits from the existing stochastic knowledge. This can be attributed to the fact that with higher levels of structural diversity, the stochastic knowledge possesses a higher value, since the request occurrence rates in regions and in time slices are less stationary due to the large temporal and spatial variance. Therefore, vehicle positions resulting from last-serviced requests significantly lose value with increasing structural diversity levels. Since the performance of the vehicle waiting strategy that is applied in the deterministic real-time control approach depends on the value of these positions, it is significantly outperformed by PROACTIVE10 for these high structural diversity settings. Note that substantial improvements can be achieved even with the smallest fleet size of eight vehicles regarding average customer inconvenience as well as late customer requests. Due to the limited resources in this case, incorrectly forecasted customer requests that lead to a misplaced vehicle are very costly.

Where there is low structural diversity present, small $DC^{min\lambda}$ values achieve the best results. This can be explained by additionally considering Table 5 that illustrates the number of generated dummy customers for different structural diversity settings. It becomes obvious that a minimum level of structural diversity is necessary to generate dummy customers of the respectively required reliability. For example, for $DC^{min\lambda} = 2.0$, a minimum structural diversity of ($RD = TD = 0.75$) is necessary. Therefore, low structural diversity settings do not allow to generate dummy customers that can be forecasted with such a high reliability. However, in scenarios with larger structural diversity, larger $DC^{min\lambda}$ values perform best for fleet sizes of 8 and 10 vehicles. Note that this effect no longer applies for the largest tested fleet size of 12 vehicles. Due to the higher quality of the stochastic knowledge that is available in higher structural diversity settings, a significant number of future requests can be reliably forecasted. By efficiently integrating these most reliable requests into the tour plan, significant reductions of customer inconvenience can be attained. However, if the fleet is enlarged to 12 vehicles and vehicles are therefore less utilized, further improvements are possible by integrating less reliable dummy customers into the tour plan. This does not apply for smaller fleet sizes since in these settings vehicles are busier and a misplacement caused by an incorrect forecast has substantial negative consequences.

All in all, the computational results indicate that the following two criteria are decisive for an efficient integration of stochastic knowledge and therefore for a successful practical application of our pro-active real-time control approach. First, a sufficient *quality of attainable stochastic knowledge in the request data* is of significant importance. It guarantees a minimum level of structural diversity that gives stochastic knowledge about future arrivals a value. Second, the *number of available resources* (i.e., the fleet size) directly influences the minimum required reliability (i.e., measured by the value of parameter $DC^{min\lambda}$) of the stochastic knowledge that leads to promising results. Specifically, with increasing fleet size this minimum required reliability decreases since the negative effects of incorrectly forecasted dummy customers diminish. Furthermore, the quadratic customer inconvenience function produces more balanced response times than linear2X. This can be mainly ascribed to the fact that, due to a considerably larger

Table 3
Performance of different deterministic real-time control approaches.

	Percentage improvements attained by	
	TIMELIMIT10 vs. ZEROTIME	ROLLING20 vs. TIMELIMIT10
Linear2X customer inconvenience		
8 vehicles	32.45%, pen: 11 vs. 175	0.22%, pen: 13 vs. 11
10 vehicles	14.33%, pen: 0 vs. 9	−0.09%, pen: 0 vs. 0
12 vehicles	9.21%, pen: 0 vs. 0	−0.15%, pen: 0 vs. 0
Quadratic customer inconvenience		
8 vehicles	51.96%, pen: 14 vs. 158	−0.79%, pen: 15 vs. 14
10 vehicles	30.65%, pen: 0 vs. 5	−0.53%, pen: 0 vs. 0
12 vehicles	19.28%, pen: 0 vs. 1	−0.56%, pen: 0 vs. 0

Note: Listed is the average improvement, pen denotes the total number of late requests.

Table 4
Percentage improvements achieved by PROACTIVE10 vs. TIMELIMIT10 (i.e., improvements attained by utilizing stochastic knowledge).

Vehicles	RegionDiversity	TimeDiversity (*TD*)				
	(*RD*)	0.00	0.25	0.50	0.75	1.00
Linear2X customer inconvenience						
8	0.00	0.08% (1.0, ∗)	↓ 1.23% (1.0, ∗)	↑ −0.18% (1.2, ∗)	↓ 1.95% (1.2, ∗)	↓ 0.26% (1.5, ∗)
	0.25	↑ −0.21% (1.0, ∗)	1.77% (1.0, ∗)	↑ 0.35% (1.0, ∗)	2.51% (1.2, ∗)	↓ 4.48% (1.5, ∗)
	0.50	1.46% (1.0, 8)	↓ 1.85% (1.2, ∗)	↓ 3.92% (1.2, ∗)	↓ 2.92% (1.5, 8)	↓ 3.85% (1.5, ∗)
	0.75	−0.55% (1.5, ∗)	1.65% (1.5, ∗)	↓ 5.06% (1.5, 7)	↓ 7.94% (1.5, 6)	↓ 10.83% (1.5, 1)
	1.00	↑ −0.33% (1.0, ∗)	↓ 2.98% (1.0, 7)	6.30% (1.5, 5)	↓ 8.92% (1.5, 5)	↓ 19.51% (2.0, 1)
10	0.00	0.51% (1.0, ∗)	1.86% (1.0, 7)	2.30% (1.2, 8)	3.22% (1.2, 8)	↓ 2.28% (1.2, ∗)
	0.25	0.79% (1.0, ∗)	2.27% (1.0, 7)	4.07% (1.0, 6)	↓ 5.21% (1.2, 4)	↑ 4.84% (1.2, 6)
	0.50	4.72% (1.0, 3)	5.37% (1.0, 3)	5.19% (1.0, 4)	6.44% (1.5, 5)	↓ 10.02% (1.5, 0)
	0.75	4.44% (1.0, 3)	6.12% (1.0, 2)	7.84% (1.2, 4)	11.14% (1.0, 2)	14.61% (1.5, 1)
	1.00	3.87% (1.0, 3)	6.75% (1.0, 2)	9.65% (1.2, 0)	14.39% (1.5, 1)	22.95% (2.0, 0)
12	0.00	1.22% (1.0, 9)	3.18% (1.0, 5)	2.04% (1.0, ∗)	4.52% (1.0, 3)	5.95% (1.0, 6)
	0.25	0.87% (1.0, ∗)	2.37% (1.0, ∗)	4.50% (1.0, 2)	6.11% (1.0, 4)	6.38% (1.2, 3)
	0.50	5.48% (1.0, 2)	6.21% (1.0, 1)	8.05% (1.0, 0)	9.04% (1.2, 0)	11.34% (1.5, 0)
	0.75	6.09% (1.0, 1)	8.46% (1.0, 0)	9.97% (1.0, 1)	15.85% (1.0, 0)	20.27% (1.2, 0)
	1.00	5.81% (1.0, 3)	9.19% (1.0, 1)	11.86% (1.2, 0)	17.99% (1.2, 0)	26.72% (1.5, 0)
Quadratic customer inconvenience						
8	0.00	1.48% (1.0, ∗)	3.02% (1.0, ∗)	↓ 3.43% (1.0, 9)	↓ 4.21% (1.2, 7)	↑ 4.57% (1.2, ∗)
	0.25	↑ 2.46% (1.0, ∗)	↓ 5.44% (1.0, 7)	↓ 6.72% (1.0, ∗)	↓ 7.17% (1.2, 9)	↓ 3.00% (1.5, ∗)
	0.50	↑ 4.14% (1.2, 8)	↑ 5.66% (1.0, 9)	↓ 10.45% (1.0, 6)	↓ 9.20% (1.2, 3)	↓ 7.06% (1.5, ∗)
	0.75	5.51% (1.0, ∗)	↓ 5.26% (1.0, 8)	↑ 9.99% (1.2, 6)	↓ 14.63% (1.5, 3)	↓ 16.75% (1.5, 2)
	1.00	8.26% (1.0, 7)	9.98% (1.2, 2)	↓ 15.16% (1.5, 3)	↓ 22.64% (1.5, 3)	↓ 29.86% (2.0, 0)
10	0.00	1.37% (1.0, ∗)	6.40% (1.0, 6)	7.79% (1.0, 6)	9.16% (1.0, 5)	5.08% (1.2, ∗)
	0.25	1.80% (1.0, ∗)	8.19% (1.0, 5)	10.13% (1.0, 2)	↓ 14.49% (1.2, 0)	↓ 10.05% (1.5, 3)
	0.50	9.08% (1.0, 3)	11.32% (1.0, 2)	15.31% (1.0, 1)	14.05% (1.0, 4)	↓ 17.89% (1.5, 1)
	0.75	9.59% (1.0, 0)	13.60% (1.0, 1)	19.88% (1.0, 1)	20.08% (1.0, 1)	27.30% (1.5, 0)
	1.00	12.20% (1.0, 4)	14.38% (1.0, 2)	19.56% (1.2, 2)	25.78% (1.5, 1)	39.33% (2.0, 0)
12	0.00	2.34% (1.0, 8)	5.02% (1.0, 8)	7.74% (1.0, 3)	13.16% (1.0, 1)	15.36% (1.0, 1)
	0.25	0.42% (1.0, ∗)	8.08% (1.0, 2)	14.04% (1.0, 2)	17.05% (1.0, 0)	15.05% (1.0, 2)
	0.50	8.72% (1.0, 2)	13.60% (1.0, 0)	16.70% (1.0, 1)	18.24% (1.0, 1)	21.64% (1.2, 1)
	0.75	10.17% (1.0, 2)	15.68% (1.0, 1)	21.01% (1.0, 0)	29.25% (1.0, 0)	37.15% (1.2, 0)
	1.00	11.48% (1.0, 2)	15.40% (1.0, 0)	24.00% (1.0, 0)	33.72% (1.0, 0)	47.24% (1.2, 0)

Note: Listed is the average improvement (best DC^{min_λ} value, # of worse instances [∗: more than 9 worse]). ↑: More total late requests than in TIMELIMIT10, ↓: less total late requests than in TIMELIMIT10.

Table 5
Number of generated dummy customers depending on the level of structural diversity.

Level of structural diversity	DC^{min_λ}				
	1.0	1.2	1.5	1.8	2.0
RD = 0.00, *TD* = 0.00	2	0	0	0	0
RD = 0.25, *TD* = 0.25	17	3	0	0	0
RD = 0.50, *TD* = 0.50	36	20	7	0	0
RD = 0.75, *TD* = 0.75	54	38	24	15	11
RD = 1.00, *TD* = 1.00	80	65	46	35	29

penalty, requests serviced late cannot be that easily compensated by other requests serviced early in the quadratic case.

7. The degree of structural diversity

In the previous section, two different dimensions of diversity were introduced. Particularly, it was shown that an efficient integration of stochastic knowledge requires a minimum degree of RegionDiversity as well as TimeDiversity. However, although this allows new significant insights into the problem structure, a general definition of structural diversity is not provided. So far, different scenarios with individual diversities are iteratively generated by linearly transforming an initial diversity setting denoted as *RD* = *TD* = 1.00. Therefore, the defined settings neither allow a classification of given request data sets nor cover all relevant characteristics. For instance, the initial setting *RD* = *TD* = 1.00 is not the maximum structural diversity that can exist in a request data set.

Such a general definition of structural diversity may allow the identification of scenarios where the integration of stochastic knowledge is reasonable. Moreover, it may provide decision support on how to efficiently customize the pro-active real-time control approach which significantly improves its practical applicability. Consequently, in what follows, we propose a first approach for determining the structural diversity of a given request data set. Specifically, an one-dimensional measure denoted as the *degree of structural diversity (dosd)* is proposed. It ranges from 0 to 1. While 0 corresponds to a request data set of no structural diversity, the value 1 defines a theoretical upper bound on the maximum structural diversity of a request data set.

In order to quantify the spatial variations in request arrivals during the considered planning horizon, the proposed *dosd* separates the planning horizon into T time periods of m minutes and tracks the changes of the barycenter of request arrivals between consecutive time periods. For each time period $t \in \{1 \ldots T\}$, the tuple (b_t^x, b_t^y) defines the geographical position of the barycenter of requests arrived in time period t according to the subarea separation depicted in Fig. 2. Moreover, n_t gives the number of requests that have occurred on the average in time period t in the request data set. Furthermore, we assume that the considered service area is divided into $x \cdot y$ subareas. In order to calculate the *dosd*, the distance between the barycenters of two consecutive time periods is computed and weighted with the number of request arrivals in the respective time period of the utilized request data set. By normalizing this weighted sum with an upper bound on the distance defined by $maxDist = \sqrt{x^2 + y^2}$, we obtain the following definition of the *dosd*:

$$dosd = \frac{\sum_{t=1}^{T-1} n_{t+1} \cdot \sqrt{\left(b_{t+1}^x - b_t^x\right)^2 + \left(b_{t+1}^y - b_t^y\right)^2}}{maxDist \cdot \sum_{t=1}^{T-1} n_{t+1}}$$

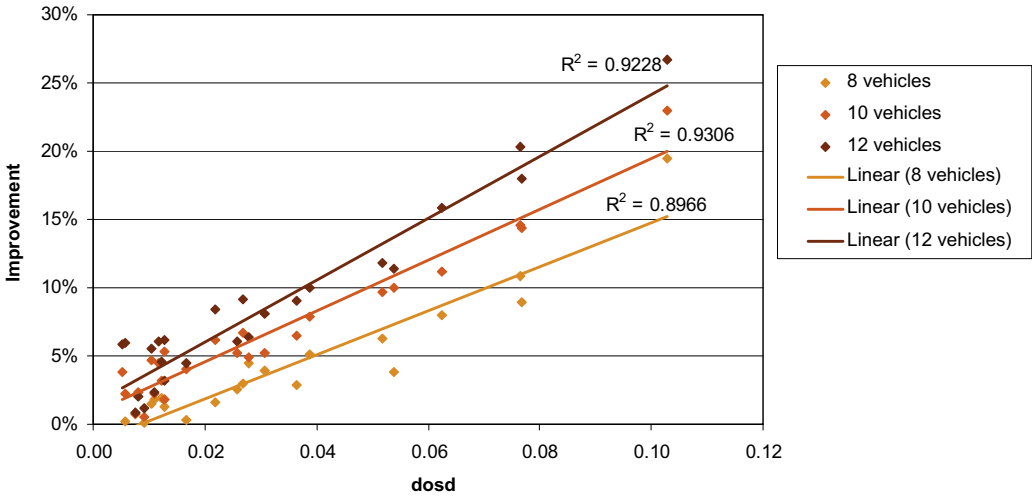

Fig. 5. Results of the degree of structural diversity (linear2X objective function).

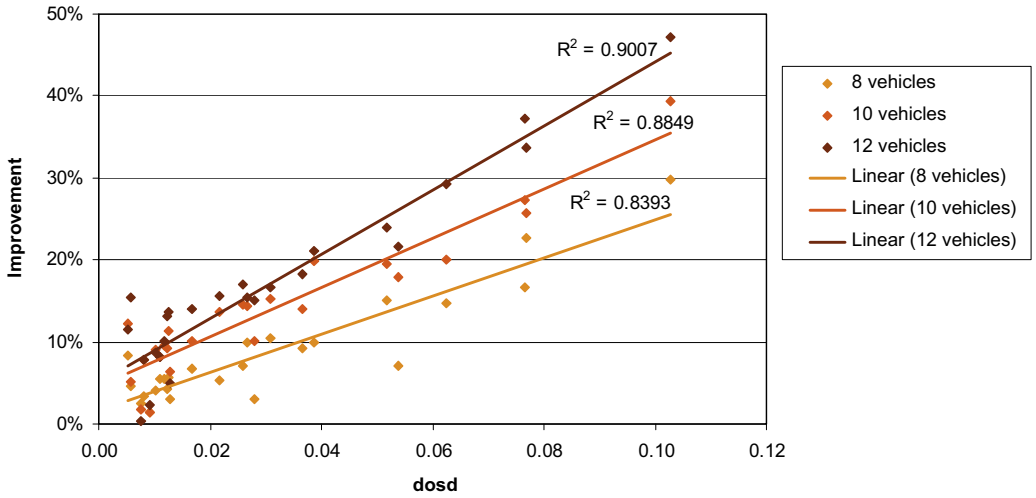

Fig. 6. Results of the degree of structural diversity (quadratic objective function).

Since in our application all vehicles start their tour at an identical depot, the first time period is used to determine starting positions of the vehicles. According to this definition, variations in request arrivals of a request data set are measured by the weighted length of an imaginary path that the majority of the vehicle fleet needs to travel in order to service all requests. Consequently, the larger the resulting distances between consecutive barycenters and the more requests occur in a time period, the more valuable is the utilization of stochastic knowledge. This is due to the fact that in such situations a deterministic approach is increasingly forced to relocate a larger number of vehicles in each time period which is likely to produce larger response times and hence increased customer inconveniences. Hence, the parameter m is approximately set to the average duration during that the vehicles have completed their relocation between two consecutive barycenters.

In what follows, we evaluate the practical applicability of the *dosd* by reference to the 25 generated request data sets introduced in Section 6.1. Specifically, we directly compare the calculated *dosd* with the average improvement rates attained by PROACTIVE10 in relation to TIMELIMIT10 (see Table 4). In order to analyze the correlation between both variables, we apply six linear regression models for each of both customer inconvenience functions combined with each of the three different fleet sizes. The results of this evaluation that are depicted in the Figs. 5 and 6 underline that the proposed *dosd* effectively identifies scenarios in which the application of stochastic knowledge is advantageous. Specifically, detailed correlation analyses using the Pearson product-moment correlation coefficient r (see Rodgers and Nicewander, 1988) provide values between 0.9161 and 0.9647, depending on the customer inconvenience function and the utilized fleet size. Accordingly, the corresponding coefficient of determination R^2 (defined by r^2) is illustrated in both figures. Consequently, a strong linear relationship between the *dosd* and the improvement that is attainable by the pro-active real-time control approach is statistically proven. Specifically, it can be concluded that a request data set with a $dosd \geqslant 0.06$ strongly indicates that the structural diversity existing in the analyzed request data set enables an efficient application of the proposed pro-active real-time control approach. Furthermore, in accordance with the results provided in Section 6.3, this is particularly true if a larger fleet size is available.

To summarize the attained promising results, it can be stated that the *dosd* provides an effective measure for classifying existing request data sets according to the application of stochastic knowledge in real-time control approaches. Specifically, depending on

the *dosd*, it can be concluded whether this is useful in the considered setting as well as how to customize the applied real-time control approach. These insights significantly improve the practical applicability of the proposed pro-active real-time control approach to real-world scenarios.

8. Conclusion and future work

This paper proposes a new pro-active real-time control approach that exploits stochastic knowledge in order to actively guide vehicles into future request-likely areas. The objective aims at minimizing customer inconvenience. Since the availability of past request information is the only requirement for generating stochastic knowledge, the approach can be applied to many real-world transportation problems.

Computational results indicate that in scenarios where the underlying request date fulfills specific criteria, the integration of derived stochastic knowledge results in significant reductions of customer inconvenience. Specifically, the pro-active real-time control approach performs best in scenarios where the available request data has a high structural diversity. Here, due to the high diversity, the derived stochastic knowledge possesses a significant value. Moreover, it has been shown that in scenarios in which the number of available resources (vehicles) is limited, it is promising to focus on stochastic knowledge of higher quality. Based on these significant insights, we finally propose a first general measure for structural diversity and statistically prove its significance.

Future work. Since the attained results are very promising and allow a general classification of past request information for the first time, future research should consider the integration of geographical information that is available in the road network into the dummy customer generation process. Specifically, an adequate handling of natural barriers (e.g., rivers or channels) require a future extension of the segment and cluster building process. Moreover, further improvements should be attained by integrating a scenario pattern generation and pattern recognition into the stochastic knowledge. Here, based on past request information, different scenario patterns are dynamically learned. While executing the process, based on the observed data, an appropriate scenario pattern is identified and applied. If such patterns exist, stochastic knowledge of higher quality can be provided.

Acknowledgment

We thank DDS Digital Data Services GmbH in Karlsruhe, Germany for generously providing us with excellent road network data.

References

Bent, R., van Hentenryck, P., 2004a. Regrets only! Online stochastic optimization under time constraints. In: Proceedings of the Nineteenth National Conference on Artificial Intelligence (AAAI-2004), San Jose, California, pp. 501–506.

Bent, R., van Hentenryck, P., 2004b. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. Operations Research 52 (6), 977–987.

Bent, R., van Hentenryck, P., 2004c. The value of consensus in online stochastic scheduling. In: Proceedings of the Fourteenth International Conference on Automated Planning and Scheduling (ICAPS-2004), Whistler, British Columbia, Canada, pp. 219–226.

Bent, R., van Hentenryck, P., 2005. Online stochastic optimization without distributions. In: Proceedings of the Fifteenth International Conference on Automated Planning and Scheduling (ICAPS-2005), Monterey, California, pp. 171–180.

Bent, R., van Hentenryck, P., 2007. Waiting and relocation strategies in online stochastic vehicle routing. In: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI 2007), Hyderabad, India, pp. 1816–1821.

Bieding, T., Görtz, S., Klose, A., 2009. On-line routing per mobile phone: a case on subsequent deliveries of newspapers. In: van Nunen, J.A.E.E., Speranza, M.G., Bertazzi, L. (Eds.), Innovations in Distribution Logistics, Lecture Notes in Economics and Mathematical Systems, vol. 619. Springer, Berlin Heidelberg, pp. 29–51.

Bock, S., 2010. Real-time control of freight forwarder transportation networks by integrating multimodal transport chains. European Journal of Operational Research 200 (3), 733–746.

Brady, M.K., Cronin Jr, J.J., 2001. Some new thoughts on conceptualizing perceived service quality: a hierarchical approach. Journal of Marketing 65 (3), 34–49.

Cordeau, J.F., Laporte, G., 2007. The dial-a-ride problem: models and algorithms. Annals of Operations Research 153 (1), 29–46.

Davis, M.M., Maggard, M.J., 1990. An analysis of customer satisfaction with waiting times in a two-stage service process. Journal of Operations Management 9 (3), 324–334.

Deutsch, P., Gailly, J.L., 1996. RFC1950: ZLIB compressed data format specification version 3.3. RFC editor United States. <http://www.ietf.org/rfc/rfc1950.txt>.

Eksioglu, B., Vural, A.V., Reisman, A., 2009. The vehicle routing problem: a taxonomic review. Computers & Industrial Engineering 57 (4), 1472–1483.

Gendreau, M., Guertin, F., Potvin, J.-Y., Taillard, E.D., 1999. Parallel tabu search for real-time vehicle routing and dispatching. Transportation Science 33 (4), 381–390.

Ghiani, G., Guerriero, F., Laporte, G., Musmanno, R., 2003. Real-time vehicle routing: solution concepts, algorithms and parallel computing strategies. European Journal of Operational Research 151 (1), 1–11.

Giaglis, G.M., Minis, I., Tatarakis, A., Zeimpekis, V., 2004. Minimizing logistics risk through real-time vehicle routing and mobile technologies. International Journal of Physical Distribution & Logistics Management 34 (9), 749–764.

Glover, F., 1989. Tabu search – Part I. ORSA Journal on Computing 1 (3), 190–206.

Glover, F., Laguna, M., 1998. Tabu Search, fourth ed. Springer, Heidelberg.

Golden, B.L., Raghavan, S., Wasil, E.A. (Eds.), 2008. The Vehicle Routing Problem: Latest Advances and New Challenges. Operations Research/Computer Science Interfaces Series, vol. 43. Springer, Heidelberg.

Hvattum, L.M., Løkketangen, A., Laporte, G., 2006. Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic. Transportation Science 40 (4), 421–438.

Hvattum, L.M., 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, J.-Y., 2000. Diversion issues in real-time vehicle dispatching. Transportation Science 34 (4), 426–438.

Ichoua, S., Gendreau, M., Potvin, J.-Y., 2006. Exploiting knowledge about future demands for real-time vehicle dispatching. Transportation Science 40 (2), 211–225.

Ichoua, S., Gendreau, M., Potvin, J.-Y., 2007. Planned route optimization for real-time vehicle routing. In: Zeimpekis, V., Giaglis, G.M., Minis, I., Tarantilis, C.D. (Eds.), Dynamic Fleet Management, Operations Research/Computer Science Interfaces Series, vol. 38. Springer Science+Business Media LLC, Boston, MA, pp. 1–18.

Kristensen, K., Kanji, G.K., Dahlgaard, J.J., 1992. On measurement of customer satisfaction. Total Quality Management & Business Excellence 3 (2), 123–128.

Kvam, P.H., Vidakovic, B., 2007. Nonparametric Statistics with Applications to Science and Engineering. Wiley Series in Probability and Statistics. Wiley, Hoboken, NJ.

Larsen, A., Madsen, O.B., Solomon, M.M., 2002. Partially dynamic vehicle routing-models and algorithms. Journal of the Operational Research Society 53 (6), 637–646.

Larsen, A., Madsen, O.B., Solomon, M.M., 2007. Classification of dynamic vehicle routing systems. In: Zeimpekis, V., Giaglis, G.M., Minis, I., Tarantilis, C.D. (Eds.), Dynamic Fleet Management, Operations Research/Computer Science Interfaces Series, vol. 38. Springer Science+Business Media LLC, Boston, MA, pp. 19–40.

MacQueen, J., 1967. Some methods for classification and analysis of multivariate observations. In: LeCam, L.M., Neyman, J. (Eds.), Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1. University of California Press, pp. 281–297.

Moon, T.K., 2005. Error Correction Coding: Mathematical Methods and Algorithms. Wiley-Interscience, Hoboken, NJ.

Parasuraman, A., Zeithaml, V.A., Berry, L.L., 1985. A conceptual model of service quality and its implications for future research. The Journal of Marketing 49 (4), 41–50.

Park, S.K., Miller, K.W., 1988. Random number generators: good ones are hard to find. Communications of the ACM 31 (10), 1192–1201.

Pollack, B.L., 2009. Linking the hierarchical service quality model to customer satisfaction and loyalty. Journal of Services Marketing 23 (1), 42–50.

Psaraftis, H.N., 1988. Dynamic vehicle routing problems. In: Golden, B.L. (Ed.), Vehicle Routing: Studies in Management Science and Systems. Elsevier Science Ltd., Amsterdam, pp. 223–248.

Psaraftis, H.N., 1995. Dynamic vehicle routing: status and prospects. Annals of Operations Research 61 (1), 143–164.

Rodgers, J.L., Nicewander, W.A., 1988. Thirteen ways to look at the correlation coefficient. The American Statistician 42 (1), 59–66.

Ross, S.M., 2010. Introduction to Probability Models, 10th ed. Academic Press, Amsterdam, Boston.

Savelsbergh, M.W.P., Sol, M., 1998. DRIVE: dynamic routing of independent vehicles. Operations Research 46 (4), 474–490.

Smith, A.K., Bolton, R.N., Wagner, J., 1999. A model of customer satisfaction with service encounters involving failure and recovery. Journal of Marketing Research 36 (3), 356–372.
Tax, S.S., Brown, S.W., Chandrashekaran, M., 1998. Customer evaluations of service complaint experiences: implications for relationship marketing. The Journal of Marketing 62 (2), 60–76.
Toth, P., Vigo, D. (Eds.), 2002. The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, Philadelphia, PA.
van de Klundert, J., Wormer, L., 2010. ASAP: the after-salesman problem. Manufacturing & Service Operations Management (12), 627–641.
van Hentenryck, P., Bent, R., 2009. Online Stochastic Combinatorial Optimization. The MIT Press.
Westphal, S., Krumke, S.O., 2008. Pruning in column generation for service vehicle dispatching. Annals of Operations Research 159 (1), 355–371.
Zeithaml, V.A., Berry, L.L., Parasuraman, A., 1996. The behavioral consequences of service quality. The Journal of Marketing 60 (2), 31–46.