Flexible Transport Process Planning in Volatile Environments and the Adaptation of a Cost-Based Objective Function

Jörn Schönberger & Herbert Kopfer

Chair of Logistics, University of Bremen, Wilhelm-Herbst-Straße 5, 28359 Bremen, Germany
{jsb, kopfer}@uni-bremen.de

1 Introduction

The formation of efficient and reliable processes to be executed in modern complex and highly volatile logistic systems requires computational support. An important prerequisite of computer based planning is the representation of the real world problem into a formal decision model. Automatic decision algorithms are then able to derive suitable solutions of such a model and one solution is selected to be executed as a process in the real world system.

For decades, researchers have defined models assuming that the planning data do not change (or that changes do not require any consideration) until the once proposed solution is completely executed. Since two decades, different and independent efforts have been made to deal with problems in which it is not possible to disregard changes of the problem data between the creation of a process and its completion.

In this contribution, we investigate a decision problem from transportation logistics that requires the update of a process under execution due to the appearance of additional problem data. Furthermore, the process planning is compromised by an unpredictably varying system load that leads to undesirable process disturbances. To overcome such situations, we propose to adjust the decision model to the current situation instead of solving the same generic problem in response to a major variation of the planning data.

Section 2 introduces the motivating decision problem. Section 3 describes a solver for single static instances of this decision problem. Section 4 is dedicated to the description of the online decision strategies to manage the variation of the problem data and Section 5 reports about the results of a comprehensive simulation study for assessing the proposed models and algorithms.

2 A Vehicle Scheduling Problem with Uncertain Demand

We present the investigated dynamic decision problem in this section. Initially, we review the relevant scientific literature (Subsection 2.1). Then, we outline the problem informally (Subsection 2.2) followed by a formalised description in terms of a decision model (Subsection 2.3). In order to allow a comprehensive assessment of the developed decision support algorithms, we propose a set of artificial but parameterisable test cases of the problem at hand (Subsection 2.4).

2.1 Literature

General layouts of dispatching systems for transportation planning tasks are proposed[1]. Real-time dispatching systems are studied in several contributions[2]. Surveys on vehicle routing and scheduling problems with incomplete planning data exist[3]. Robust planning is defined as the generation of plans that maintain their high or even optimal quality after subsequent modifications[4]. Flexible planning refers to the generation of plans whose quality does not significantly decrease after the execution of algorithmic re-scheduling and alterations of the so far used plans[5]. Robust transport scheduling approaches exploiting explicit probability distributions of expected future demand are investigated by seberal researchers[6]. Flexible planning approaches typically solve online decision problems. Krumke[7] provides a survey of online vehicle routing and scheduling problems. Fleischmann et al.[8], Gutenschwager et al.[9] as well as Savelsbergh and Sol[10] tackle real-world applications in this research field. The adaptation of the underlying formal optimisation model in order to map changes in the considered real world problem follow two directions: the adaptation of the search space definition is proposed[11] and the impacts of different weights for objective function components are investigated[12].

2.2 Informal Problem Description

The problem we are investigating in this contribution generalises the common vehicle routing problem with time windows in three aspects.

Soft Time Windows. Lateness at a customer site is possible but causes penalty costs. The compensation amount proportionally increases with the delay but is limited to a certain extremely high maximal amount for each request. Although a particular

[1] Gayalis and Tatsiopoulos (2004)
[2] Ghiani et al. (2003); Fleischmann et al. (2004); Gutenschwager et al. (2004); Séguin et al. (1997)
[3] Gendreau and Potvin (1998); Psaraftis (1995)
[4] Jensen (2001)
[5] Jensen (2001)
[6] Bianchi et al. (2005); Jaillet (1988)
[7] Krumke (2001)
[8] Fleischmann et al. (2004)
[9] Gutenschwager et al. (2004)
[10] Savelsbergh and Sol (1998)
[11] Schönberger and Kopfer (2007)
[12] Gutenschwager et al. (2003)

request may be late, p^{target} percent of the f_t requests completed in $[t-t^-; t]$ and expected to be completed in $[t; t+t^+]$ must be in time. Let $\tilde{f}_t$ be the number of the requests completed timely within the last t^- time units and let $\hat{f}_t$ be the number of punctually scheduled requests within the next t^+ time units, then $p_t := \frac{\tilde{f}_t + \hat{f}_t}{f_t} \geq p^{target}$ is postulated and p_t is taken as a measure for the reliability of the service.

Subcontracting. Each customer request is allowed to be subcontracted. Logistics service providers (LSP) are paid for the fulfilment of selected requests. An LSP receives a certain amount F_r for this service but ensures that the request r is fulfilled within the specified time window. If a request has been subcontracted then this decision cannot be revised.

Uncertain Demand. Only a subset of all requests is known to the planning authority at the time when the subcontracting is chosen and the routes for the own vehicles are generated. The planning authority decides about subcontracting or self-fulfilment as soon as additional requests become known. The demand, expressed by incoming requests, varies significantly and unpredictably over time.

We refer to the above decision problem as the vehicle scheduling problem with time windows and uncertain demand (VSPTWUD).

Let $\mathcal{R}(t_i)$ be the collection of all requests known but not completed at time t_i. A transportation plan TP_i describes how the known requests are fulfilled. The fulfilment mode is indicated as well as the information how the own vehicles, forming the fleet $\mathcal{V}$, should behave in order to execute those requests that have not been subcontracted (routes). Since additional requests are released at the (ex ante unknown) time t_i, the so far valid transportation plan TP_{i-1} (generated at time t_{i-1}) becomes void and requires an update to a new transportation plan TP_i considering also the additional requests released at time t_i. This plan TP_i is followed until at time t_{i+1} a new update becomes necessary and TP_i is replaced by TP_{i+1}. Actually, a sequence $TP_0, TP_1, TP2, \ldots$ of transportation plans is generated and each single transportation plan is executed until the necessity for serving additional requests corrupts the plan execution.

The primary goal of the update of TP_{i-1} is to include the recently released requests into the transport processes. Thereby, it is aimed at keeping the costs for the execution of the additional requests as low as possible but, on the other hand, to provide a sufficiently high reliability within the request fulfilment.

In order to ensure that the updated transportation plan is realizable, the following conditions (a)-(e) have to be considered for the transportation plan update.

(a) Exactly one path is selected for each vehicle $v \in \mathcal{V}$.

(b) Each request $r \in \mathcal{R}(t_i)$ is fulfilled by a vehicle $v \in \mathcal{V}$ or it is subcontracted.

(c) A once subcontracted request cannot be contained in any of the selected paths.

(d) Once a request r has been assigned to state (S) it cannot be re-assigned to another vehicle or LSP as determined in TP_{i-1}.

(e) The punctuality of the transportation plan is at least p^{target}.

The costs for serving the additional requests should be as minimal as possible.

2.3 Problem Modelling

A single request r belongs consecutively to different states. Initially r is known but not scheduled (F=fresh). Then, r is assigned to an own vehicle (I=internal fulfilment) or subcontracted (E=external fulfilment). If the working at the corresponding customer site has already been started but not yet been finished r is assigned to the state (S=service).

The set $R^+(t_i)$ is composed of additional requests released at time t_i. Requests completed after the last transportation plan update at time t_{i-1} are stored in the set $R^C(t_{i-1}, t_i)$. The new request stock $R(t_i)$ is determined by $R(t_i) := R(t_{i-1}) \cup R^+(t_i) \setminus R^C(t_{i-1}, t_i)$.

A transportation plan revision at time t_i starts with the state update for the requests contained in $R(t_i)$. The state (F) is assigned to all recently added requests. Requests contained in $R^C(t_i)$, whose on-site execution has been started after t_{i-1} but not completed at time t_i receive the state (S). After the update decisions have been made the state of an (I)-labelled request is updated to (E) if it has been decided to be out-sourced. Finally, all (F)-labels of subcontracted requests are replaced by (E)-labels or (I)-labels depending on the selected fulfilment mode.

In order to update the so far followed transportation plan TP_{i-1} the fulfilment mode for the requests contained in $\mathcal{R}(t_i)$ have been determined and a path to be followed has to be assigned to each of the vehicles of the own fleet $\mathcal{V}$ (representing the processes to be followed by the vehicles).

Let $M(r)$ denote the current state of request $r \in \mathcal{R}(t_i)$. The set $\mathcal{R}^X(t_i)$ contains all currently executed requests r with $M(r) = X$.

If request r has been contained in a path of an own vehicle in the so far valid transportation plan TP_{i-1} then $v(r) \in \mathcal{V}$ denotes the corresponding vehicle. Otherwise, we set $v(r) = -1$.

All vehicles start at time $t_0 = 0$ at a designated depot D, fulfil their operations according to the subsequently updated transportation plans and return to the depot at the end of the considered time interval. The location of vehicle v at time t_i is stored in $s^v_{t_i}$ and a feasible path $p = (p_1, p_2, \dots, p_{n_p})$ for vehicle v holds for a) $p_1 = s^v_{t_i}$, b) $p_{n_p} = D$ and c) $p_i \in \{s^v_{t_i}, D\} \cup \mathcal{R}(t_i)$. All paths end in the depot D. If a vehicle is waiting at the depot location, then the empty path D, D is feasible (and represents the non-consideration of this vehicle in the planned operations). The insertion of additional customer site visits defers the return to D but if the vehicles are idle (because no additional requests require a service) then the vehicles should return to the depot and wait there for further instructions.

All paths, which are feasible for vehicle v at time t_i, are collected in the set $\mathcal{P}_v(t_i)$ and all paths feasible for at least one vehicle at time t_i are consolidated in the set $\mathcal{P}(t_i) := \bigcup_{v \in \mathcal{V}} \mathcal{P}_v(t_i)$.

If the request r is served by path p then the parameter a_{rp} is set to 1, otherwise it is set to 0.

In order to evaluate the decisions about the fulfilment mode and the selected routes, we calculate the costs associated to each path p. The travel costs associated with path p are denoted as $C^1(p)$. We calculate the penalties associated with a path as follows. At first, we determine the delay at each served customer site. A linear function maps the delay to an amount of compensation payments that increases with growing delays. As soon as the delay climbs over a given threshold, this amount is not further increased. Finally, the calculated sum of penalty payments is stored in $C^2(p)$. The costs for the subcontracting of a certain request r are labelled as $C^3(r)$.

We deploy two families of binary decision variables. Let $x_{pv} = 1$ if and only if path $p \in \mathcal{P}_v(t)$ is selected for vehicle $v \in V$ and let $y_r = 1$ if and only if request $r \in \mathcal{R}(t_i)$ is subcontracted.

The set partition model of the short-term decision problem instance $SP(t_i)$ to be solved at time t_i is given by (1)-(5). The selected solution of this model becomes the updated transportation plan TP_i.

$$\sum_{p \in \mathcal{P}(t_i)} \sum_{v \in \mathcal{V}} (C^1(p) + C^2(p)) x_{pv} + \sum_{r \in \mathcal{R}(t_i)} C^3(r) y_r \quad \rightarrow \quad \min \tag{1}$$

$$\sum_{p \in \mathcal{P}_v(t_i)} x_{pv} = 1 \qquad \forall v \in \mathcal{V} \tag{2a}$$

$$x_{pv} = 0 \qquad \forall p \notin \mathcal{P}_v(t_i), v \in \mathcal{V} \tag{2b}$$

$$y_r + \sum_{p \in \mathcal{P}(t_i)} \sum_{v \in \mathcal{V}} a_{rp} x_{pv} = 1 \qquad \forall r \in \mathcal{R}(t_i) \tag{3}$$

$$y_r = 1 \qquad \forall r \in \mathcal{R}^E(t_i) \tag{4}$$

$$\sum_{p \in \mathcal{P}_{v(r)}(t_i)} a_{rp} x_{pv(r)} = 1 \qquad \forall r \in \mathcal{R}^S(t_i) \tag{5}$$

The two restrictions (2a) and (2b) correspond to condition (a), (3) ensures the consideration of (b), (4) refers to (c) and (5) guarantees that (d) is met. In order to support the fulfilment of condition (e), we consider the penalty costs for late arrivals within the objective function (1) as well as the travel and subcontracting costs. The NP-hardness of this model is obvious since the Travelling Salesman Problem is a special case of this model.

2.4 Construction of Artificial Test Cases

Two different kinds of routing scenarios with successively arriving requests are mentioned in the scientific literature. The amount of additional demand remains unchanged over time in a *stable scenario*[13]. Resources can be adapted on the longer term to enable the

[13] Lackner (2004); Mitrović-Minić et al. (2004); Pankratz (2002)

fulfilment of the complete demand in time. In case that the amount of additional demand varies significantly over a specific time interval, the dispatching unit has to manage demand peaks that might violate capacity constraints. In such a *peak scenario* it is nearly impossible to adapt the available resources in advance[14]. A parameterisation and/or classification for scientific analysis purposes is hardly realizable for these instances, which typically correspond to real world scenarios. For this reason, we have decided to define a new set of artificial but parameterisable test cases.

We first generate a balanced stream of incoming customer demands over the complete observation period $[0; T_{max}]$. Therefore, n_0 requests are drawn randomly from the Solomon's[15] 100-customer-vehicle routing problem with time windows instance P at time $t_{rel} = 0$. Then, the request release time is updated to $t_{rel} := t_{rel} + \Delta t$ and for this new release time, n_0 customer requests are drawn from P at random again. However, for each of the recently selected requests r, the release time is set to t_{rel}. The original service time window $[e_r; l_r]$ of r is replaced by $[t_{rel} + e_r; t_{rel} + l_r]$. Then, t_{rel} is increased by Δt again and additional requests are generated similarly as long as $t_{rel} \leq T_{max}$. A second stream of demand is generated in order to achieve a peak of demand. Again, we iteratively increase the release time t_{rel} by Δt starting at $t_{rel} = 0$. As long as $t_{rel} \leq t_{start}^{peak}$ is met no additional demand occurs. In case that $t_{rel} \in [t_{start}^{peak}; t_{start}^{peak} + d_{peak}]$, n_1 additional requests, drawn randomly from P, are selected to be released at t_{rel}. Again, the original service time window is shifted by t_{rel}. No requests are specified anymore within the second stream as soon as $t_{rel} > t_{start}^{peak} + d_{peak}$. Both streams are then overlaid so that during the period $[t_{start}^{peak}; t_{start}^{peak} + d_{peak}]$ a higher number of requests appears.

All vehicles specified within the instance P can be used. In order to determine a competitive and comparable tariff for calculating the LSP fare F_r associated to a request r, we desist from capacity constraints and set the capacity usage of each request to zero. We multiply the Euclidian distance d_r between the depot of the LSPs, situated at location $(65, 65)$, and the customer site associated to r with a normalising factor ν_r. A subcontracting of r costs $F_r := d_r \cdot \nu_r$ monetary units. We consult the best-known solution $\mathcal{S}(P)$ of P found in the literature in order to calculate ν_r. The vehicle v_r serves r according to $\mathcal{S}(P)$ and $l^{demanded}$ denotes the sum of the Euclidian distances (the demanded distances) between the depot and the customer sites of all requests served by v_r in this solution proposal. The normalising factor assigned to request r is now set to $\nu_r := \alpha \cdot \frac{l^{demanded}}{l^{travelled}}$, where $l^{travelled}$ denotes the route length of vehicle v_r [16]. Scenarios with different tariff levels are generated by modifying the factor α. If $\alpha << 1$ then subcontracting is cheaper than the self fulfilment, in case that $\alpha \approx 1$ both fulfilment modes have comparable costs but if $\alpha >> 1$ then the self completion mode is cheaper.

Each scenario is described by the 5-tuple $(P, d_{peak}, n_0, n_1, \alpha)$. In this investigation, we use the four Solomon cases $P \in \{R103, R104, R107, R108\}$ to generate request sets with tariff levels $\alpha \in \{1.0, 1.25, 1.5, 1.75, 2, 3\}$. Furthermore, it is $n_0 = 50$, $n_1 = 100$ and $\Delta t = 100$ time units. The peak duration is fixed to $d_{peak} = 200$ time units starting at

[14] Gutenschwager et al. (2004); Hiller et al. (2006); Fleischmann et al. (2004)
[15] Solomon (1987)
[16] Schönberger (2005)

$t_{start}^{peak} = 1500$ time units. Finally, the total observation period is $T_{max} = 5000$ time units.

3 Memetic Algorithm Schedule Generation

We use a Memetic Algorithm (MA) realizing a hybrid search strategy consisting of a global genetic search space sampling and a local 2-opt improvement procedure for solving the scheduling model instances $SP(t_0), SP(t_1), \ldots$ of the online decision problem introduced in 2.3

The genetic search uses a $\mu + \lambda$-population model evolved by the application of the PPSX-crossover-operator [17] and a mutation operator that a) moves arbitrarily selected operations between LSPs and the own fleet routes, b) shifts requests between selected routes of own vehicles and c) reverses the visiting order of randomly chosen subsequences of arbitrarily selected routes.

The construction of the initial population is generated using the Push Forward Insertion Heuristic[18]. One half of the initial set of solution proposals is generated by deploying the heuristic followed by some random proposal modifications and the other half is generated purely at random without applying any biasing procedure. The evolution process is stopped dynamically if the average fitness of the evolved population does not improve for 10 generations.

Every time a new decision model instance $SP(t_i)$ is arriving the MA is re-started to solve the model of the recent instance. Initial experiments, in which parts of the final population of the last instance solved are used to seed the initial population of the recent instance, failed because the recent population converges too rapidly on a too bad level even if the crossover and mutation probability are determined adaptively. An analysis of the population development has shown that the significantly varied decision situation requires the re-initialisation of the genetic material so that the new decision aspects are considered explicitly. For this reason, a complete new initial population is formed using the seeding approach described above.

4 Online Optimisation Strategies

This section is about the description of decision strategies for the management of the transportation plans in the evolving scenario. First, we describe a state-of-the-art re-optimisation procedure that solves a sequence of the same generic decision model (Subsection 4.1). Second, we propose an extension of this approach in which the decision model is adapted to the current system performance (Subsection 4.2).

4.1 Repeated Cost Minimisation

As soon as additional requests become known and the update of TP_{i-1} to TP_i is necessary, the generic decision model (1)-(5) is stated. Then, the MA is started to derive a

[17] Schönberger (2005)
[18] Solomon (1987)

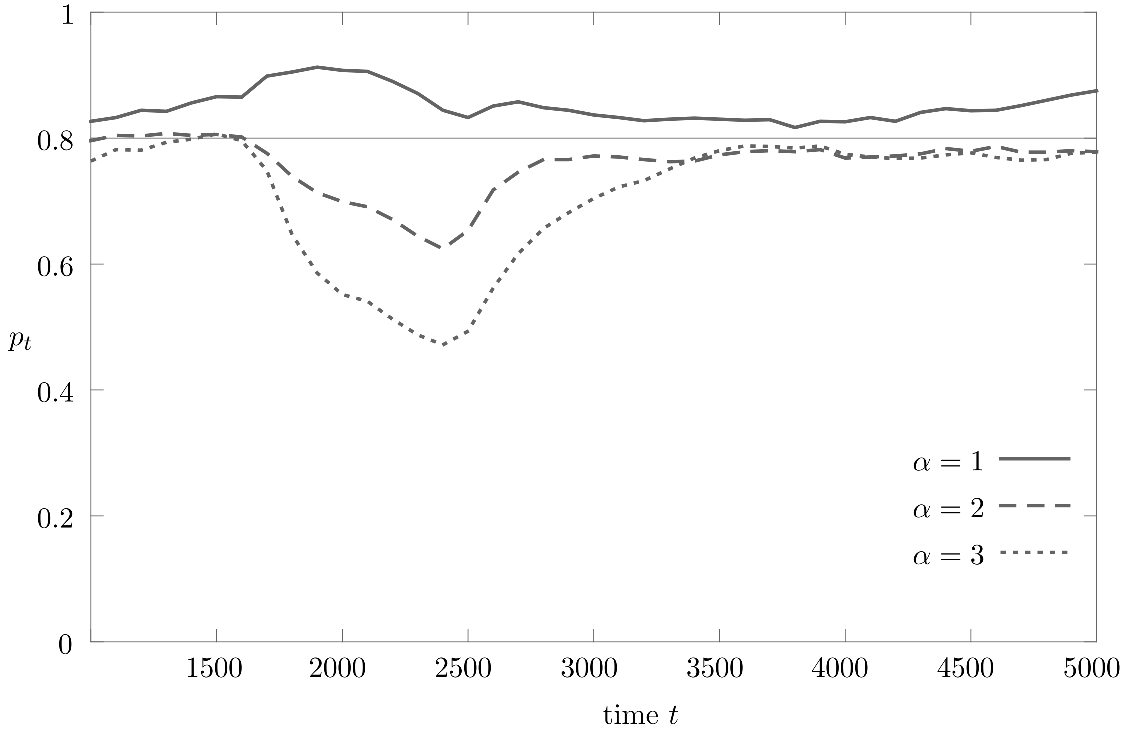

Figure 1: Development of the punctuality p_t over time

solution of the current instance $SP(t_i)$of this model.

Each of the $|\{R103, R104, R107, R108\}| \cdot |\{1, 1.25, 1.5, 1.75, 2, 3\}|$=24 scenarios is simulated using the aforementioned re-optimisation strategy. Three independent runs are performed for each scenario. The averagely observed results are reported in the remainder of this subsection. Here, the target punctuality is defined as $p^{target} = 0.8$ represented by the continuous horizontal graph in Fig. 1. We set $t^+ = t^- = 500$ so that the recent punctuality p_t is calculated at time t with respect to the moving time window $[t - 500, t + 500]$. A piece-wise linear penalty function is deployed, which is 0 for no delays and which increases proportionally up to a maximal value of 25 money units for delays longer than the threshold delay 100 time units.

The development of the punctuality p_t during the experiments shown in Fig. 1 complies with the reliability requirements only if fair tariffs ($\alpha = 1$) are available. In this case, the punctuality varies between 0.8 and 0.9 throughout the complete simulation experiment. As soon as the subcontracting tariffs are enlarged, the penalisation of too late arrivals within the objective function (1) is not sufficient any more and the punctuality sinks below 65% ($\alpha = 2$) and 50% ($\alpha = 3$) respectively. Furthermore, if $\alpha = 2$ then p_t does never climb again over p^{target} and p_t remains below 80% for the remaining observation time.

We have recorded the percentage σ_t of subcontracted requests within the experiments

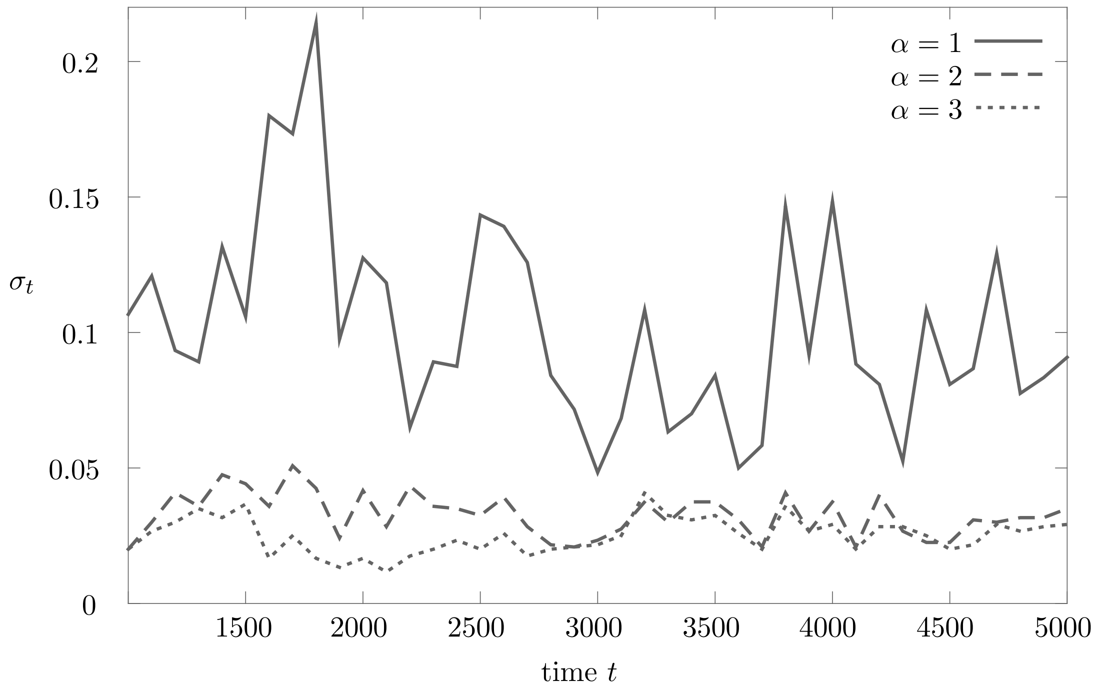

Figure 2: Development of the externalization portion σ_t

in the moving time window $[t - t^{-}; t + t^{+}]$ in order to find out the reason(s) for the poor performance of the cost-based re-optimisation strategy. As it can be depicted from Fig. 2, subcontracting is only used to a larger extend if the subcontracting tariffs are comparable with the self-fulfilment costs ($\alpha = 1$). In such a case, σ_t climbs over 20% as an immediate response to the increased demand just after $t = 1500$. If the subcontracting tariffs are increased, no more than 5% of the current request portfolio are out-sourced because the strive for a least cost new transportation plan detects that it is cheaper to accept further penalty charges than to subcontract additional requests.

4.2 Situation-Based Adaptation of the Objective Function

Two states of the dynamic problem have to be distinguished depending on the current service punctuality p_t at a particular time t. In a *high quality state (HQ)* the requirement for the least punctuality is fulfilled ($p_t \geq p^{target}$). The minimisation of the transportation plan costs is claimed and produces transportation plans of sufficient reliability. In contrast, in a *low quality state (LQ)* the required punctuality is not attained anymore ($p_t \leq p^{target}$). Updating the transportation plan by considering primarily the costs causes further punctuality decreases. Consequently, the model proposed in Subsection 2.3 is appropriate only for the transportation plan update in HQ states but in LQ states this model is not adequate and requires an adjustment so that the primary goal is to re-achieve the punctuality p^{target}.

The original online decision making framework [19] is neither capable to detect changes in the considered problem that requires a new model nor is it equipped to implement these model modifications. In order to overcome these deficiencies, we propose three extensions of the online decision making framework.

Preparations. In order to decide, whether the system's current performance corresponds to an HQ or LQ state, we first specify the intended system development starting from the current time t_i. Therefore, we select N indicators that map the performance of the considered logistic system at a time t into the N-tuple $(i_1(t), \ldots, i_N(t))$ of real values (the system's state at time t). Let Im_u denote the set of possible values forthe indicator $i_u : t \mapsto i_u(t)$. Furthermore, the set $\mathcal{F}(t) \subseteq Im_1 \times \ldots \times Im_N$ exactly contains all those system states that are desired. The set $\mathcal{D}(t_i) := [t_i; \infty[\times\mathcal{F}(t_i)$ contains all feasible future system states. It is called the *System Development Corridor at time* t_i.

The system development corridor for the problem introduced in Section 2 is defined as follows. We use the only indicator p_t mapping the current punctuality into a real value, $Im_1 := [p^{target}; 1]$ and set $\mathcal{F}(t) := [p^{target} + 0.05; 1]$. The corridor $\mathcal{D}(t_i)$ is then given by $\mathcal{D}(t_i) := [t_i; \infty[\times[p^{target} + 0.05; 1]$. Since the system development corridor is already left before p_t reaches p^{target}, there is time to establish countermeasures that lead to a re-increase of p_t.

Adjustment Intensity Determination. A transportation plan update becomes necessary at time t_i. At first, the current system performance $x(t_i) := (i_1(t_i), \ldots, i_N(t_i))$ is determined by reading the current values of the indicators. Then, it is checked whether $(t_i, x(t_i))$ belongs to $\mathcal{D}(t_i)$ (HQ state). Therefore, the continuous real-valued function h is evaluated (modification intensity $h(t_i, x(t_i), \mathcal{D}(t_i))$. This function is 0 as long as $(t_i, x(t_i)) \in \mathcal{D}(t_i)$ but increases monotonously if $(t_i, x(t_i))$ moves away from $\mathcal{D}(t_i)$.

We use the piece-wise linear function h in this contribution that is 0 as long as $p_t \geq p^{target} + 0.05$ (HQ state), $h(p_t) = 1$if $p_t \leq p^{target} - 0.05$ (LQ state) and which decreases monotonously from 1 down to 0 if p_t increases from 0.75 up to 0.85 (transition state).

Model Adjustment Instantiations. The function H maps the modification intensity $h(t_i, x(t_i), \mathcal{D}(t_i))$ to a set of modifications that transforms the so far used decision model $SP(t_{i-1})$ into the updated decision model $SP(t_i)$. If $h(t_i, x(t_i), \mathcal{D}(t_i)) = 0$ then $H(SP(t_{i-1}), h(t_i, x(t_i), \mathcal{D}(t_i))) := \emptyset$ (no model modifications are necessary if the system development corridor is not left). Finally, the just defined modifications are established and the new decision model $SP(t_i)$ is solved.

The model $SP(t_i)$ to be solved at time t_i is then given by the constraint set (2a)-(5) together with the objective function defined by (6).

$$\sum_{p \in \mathcal{P}(t_i)} \sum_{v \in \mathcal{V}} f(t_i) \cdot (C^1(p) + C^2(p)) x_{pv} + \sum_{r \in \mathcal{R}(t_i)} C^3(r) y_r \quad \rightarrow \quad \min \tag{6}$$

The coefficient $f(t_i)$ is adjusted for every decision model $SP(t_i)$. It is defined by

[19] Krumke (2001)

$$f(t_i) = \begin{cases} 1, & i = 0 \\ 1 + \alpha \cdot h(t_i, x(t_i), \mathcal{D}(t_i)) & i \geq 1 \end{cases}$$

Thus, in this context, we use the function h that determines the current value of $f(t_i)$. If the current punctuality p_{t_i} is larger than $p^{target} + 0.05$ than $f(t_i)$ is 1 and the generic model as introduced in Section 2 is used for the transportation plan update. In case that $p_{t_i} \leq p^{target} - 0.05$ it is $f(t_i) = 1 + \alpha$ and subcontracting a request is identified by the solver via the objective function to be cheaper than the self-fulfilment with respect to the currently used objective function (6). The intention of varying the relative weight of the costs of the two fulfilment modes is to make the subcontracting mode more attractive by lifting the costs of the self fulfillment mode. Therefore, it is necessary to enlarge both costs drivers (travel costs and penalties) associated with the selfulfilment of a requests. Consequently, the MA will prefer those transportation plans that come along with a higher number of subcontracted requests. Initial experiments have shown that a sole adaptation of the penalty function leads to reasonable results only in some special situations.

The function H used to implement the model adjustments is realised by the update of the weighting coefficient from $f(t_{i-1})$ to $f(t_i)$. Since the re-definition of the coefficient affects the search trajectory heading of the solving algorithm, we call this approach the Search Direction Adaptation (SDAD) strategy. The approach without model adaptation proposed in Subsection 4.1 is referred to as NONE (intervention) strategy.

5 Computational Experiments

In this section, we report about the results of the numerical simulation experiments. Subsection 5.1 describes the setup of the experiments. In Subsection 5.2, the observed results are shown and an interpretation is given.

5.1 Layout of the Experimental Field

Again, the target punctuality is set to $p^{target} = 0.8$ to be achieved in the moving time window $[t - 500, t + 500]$ in order to compare NONE and SDAD results directly.

Each of the $|\{R103, R104, R107, R108\}| \times |\{1, 1.25, 1.5, 1.75, 2, 3\}| = 4 \cdot 6 = 24$ scenarios is simulated three times leading overall to 72 simulation experiments that have been performed. Here, we report the average results observed for each scenario.

For analysing the impacts of the objective function adaptation with respect to the tariff level α, we first calculate p_t's maximal decrease $\delta(\alpha) := \frac{\min_{t \geq 1500}\{p_t(\alpha)\}}{p_{1000}(\alpha)}$ after the demand peak's start.

Let T_α^{below} denote the first time in which p^{target} is not achieved and T_α^{heal} refers to the time in which a reliable state is finally re-achieved. We define $\pi(\alpha) := \frac{T_\alpha^{heal} - T_\alpha^{below}}{4000}$ as the percentage of low quality states within the observation interval $[1000, 5000]$.

Furthermore, we have recorded the number of waiting requests $QL_t(\alpha)$. We define $\phi(\alpha) := \max_{t \geq 1500}\{QL_t(\alpha)\}$ to be the maximal observed number of known but not fulfilled requests.

Beside the above effects on the generated processes, we have recorded the associated costs. Let $C_\alpha(t)$ denote the cumulated overall costs realized up to time t. In order to quantify the impacts of tariff level enlargements, we calculate the relative growth of costs $c(\alpha) := \frac{C_\alpha(5000)}{C_1(5000)}$. Similarly, we calculate the relative growth of the travel costs ($c^{INT}(\alpha)$), of the subcontracting costs ($c^{EXT}(\alpha)$) as well as of the penalty costs ($c^{PEN}(\alpha)$).

The contribution of the travel costs to the overall costs is defined as $m^{INT}(\alpha) := \frac{C_\alpha^{INT}(5000)}{C_\alpha(5000)}$ (the portion $m^{EXT}(\alpha)$ of the subcontracting costs as well as the portion $m^{PEN}(\alpha)$ of the penalties are determined similarly).

In all case, we report the costs associated with the processes (travel costs, penalties and subcontracting fees) and not the objective function values, which are only used to guide the search algorithm through the search space.

5.2 Presentation and Interpretation of Numerical Results

The averagely observed values for the adaptation factor $f(t)$ recorded in experiments with different tariff levels α, are represented in Fig. 3. However, the qualitative development of f_t throughout the simulation remains unaffected for different α-values. Initially, $f(t)$ waits and remains unchanged as long as $t < 1700$ (stand-by phase). Next, an intervention becomes necessary and $f(t)$ increases until it reaches its maximal value at $t = 2400$ (intervention phase). This phase is followed by a descending phase ($2400 \leq t \leq 2500$). After the demand peak impacts have been managed, $f(t)$ oscillates nervously around its initial value ($2500 < t \leq 4200$) before it re-enters a stable waiting phase ($t > 4200$). The observed results prove that the proposed control function $h(\cdot)$ generates a proper response to the changes of the system's input independently from the tariff level α.

If the difference between the costs for the two fulfilment modes increases (represented by ascending α-values) then the level of $f(t)$ increases as well from 1 ($\alpha = 1$) up to ≈ 2 ($\alpha = 3$). Additionally, the maximal values achieved at the end of the intervention phase grow up as long as the tariff level α is enlarged.

To study the impacts of the different strategies NONE and SDAD on the punctuality in case that the tariff level α is varied, we compare the relative minimal punctuality level $\delta(\alpha)$ averagely observed in the performed experiments (Tab. 1). In general, the punctuality indicator $\delta(\alpha)$ reduces with increasing tariffs level α independently from the application of SDAD. This is mainly caused by the declining attractiveness of the reliable subcontracting mode if α is enlarged. However, the observed results clearly indicate that a context-based model adjustment supports the preservation of a high punctuality level. In the NONE-experiment the relative minimal punctuality level $\delta(\alpha)$ decreases from 98.8% ($\alpha = 1$) down to 61.8% ($\alpha = 3$). If the SDAD-strategy is applied then $\delta(\alpha)$ remains above 93% for all analysed tariff levels α. It falls from 99% ($\alpha = 1$) down to 94.3% ($\alpha = 3$). The observed results allow the conclusion that the search function

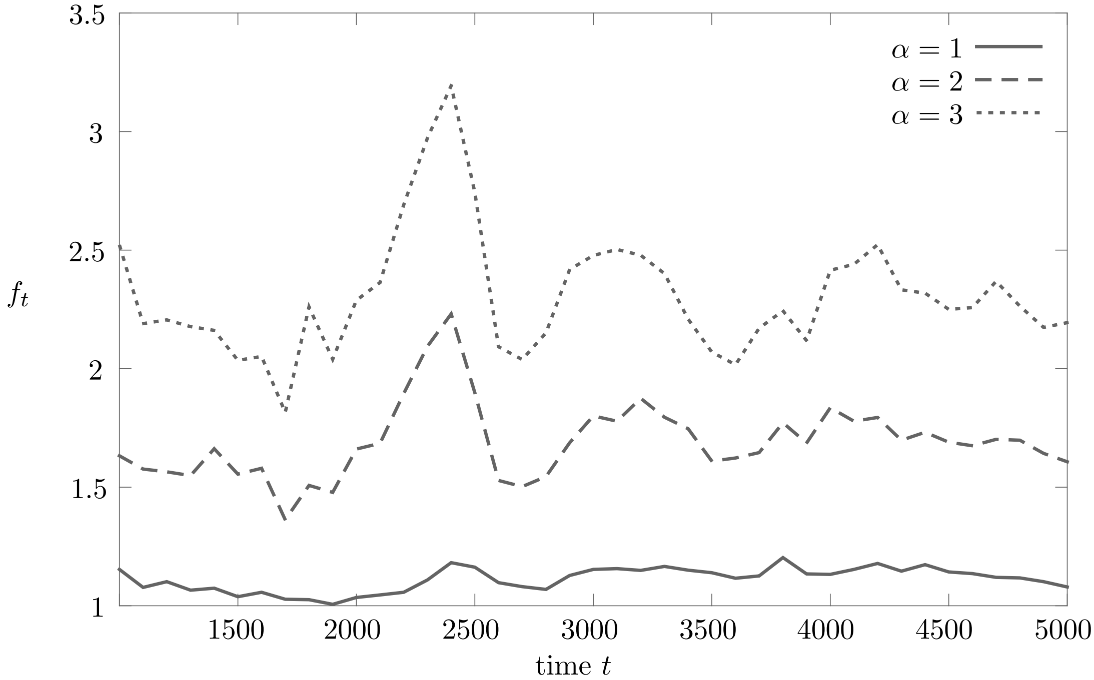

Figure 3: Averaged adaptation factors $f(t)$ observed during the simulation

adaptation is appropriate to maintain a high punctuality rate independently from the currently valid tariff level.

The results compiled in Tab. 1 show the severeness of the demand peak. In addition, Tab. 2 shows the percentage $\pi(\alpha)$ of the relevant part $[1000, 5000]$ of the observation interval in which the intended target punctuality $p^{target} = 0.8$ is not finally re-achieved (Tab. 2). If no model adaptation is carried out then the length of the LQ state increases from $\pi(1) = 0\%$ up to $\pi(3) = 97.5\%$. In case that SDAD is applied, LQ-states are completely prevented for $\alpha \leq 1.5$ and for larger tariff levels only small percentages are observed ($\pi(1.75) = 5\%, \pi(2) = 2.5\%$) except for the highest tariff level ($\pi(3) = 50\%$). We learn from this observation that SDAD is able to reduce the duration of LQ-states

Table 1: Maximal punctuality decrease $\delta(\alpha)$

	α					
	1	1.25	1.5	1.75	2	3
NONE	98.8%	97.9%	92.4%	86.8%	78.4%	61.8%
SDAD	99.0%	95.8%	94.5%	95.5%	93.0%	94.3%

Table 2: Percentage of LQ-states $\pi(\alpha)$

	α					
	1	1.25	1.5	1.75	2	3
NONE	–	60.0%	70.0%	97.5%	82.5%	97.5%
SDAD	–	–	–	5.0%	2.5%	50.0%

Table 3: Maximal percentage of subcontracted requests $\sigma(\alpha)$

	α					
	1	1.25	1.5	1.75	2	3
NONE	21.4%	15.6%	10.0%	5.8%	5.1%	4.0%
SDAD	23.0%	18.5%	19.2%	17.4%	15.0%	14.5%

significantly.

The impacts of the decision model adaptation on the fulfilment mode selection are summarised in the values presented in Tab. 3. Without any model adjustment, the highest observed subcontracting percentage decreases with increasing tariff level α from $\sigma(1) = 21.4\%$ down to $\sigma(3) = 4.0\%$ which is a reduction by more than 80%. The application of SDAD leads to significantly different results. At first, the observed externalisation rate is higher for every tariff level α and, secondly, the loss of $\sigma(\alpha)$ after increasing α from 1 up to 3 is only ≈37%. We conclude that the adaptation of the objective function makes subcontracting more attractive by re-weighting the decision relevant costs.

The application of SDAD leads to a reduced number $v(t)$ of averagely deployed own vehicles $v \in V_t$ (Fig. 4) if the tariff level is quite incomparable ($\alpha = 3$). If no model adaptation is applied (NONE-experiment) then averagely 10 vehicles from the own fleet are scheduled as long as no demand peak occurs. The application of SDAD leads to a reduced number of 8 deployed vehicles in off-peak times. Furthermore, the maximal number of deployed own vehicles in the NONE experiment is 25, but only 22 vehicles are simultaneously scheduled if SDAD is applied. Both strategies NONE and SDAD lead to an immediate increase of deployed vehicles in response to the demand peak starting at time 1500. However, this number declines only slowly in the NONE experiment while in the SDAD experiment it reduces to the off-peak average quite faster.

The significantly higher percentage of subcontracted requests leads to a reduction of the length $\phi(\alpha)$ of the queue built by those requests that already have been released and scheduled but not yet been completed (Tab. 4). Generally, $\phi(\alpha)$ increases if the tariff level α is raised. In the NONE experiment $\phi(1) = 195.0$ and the queue length

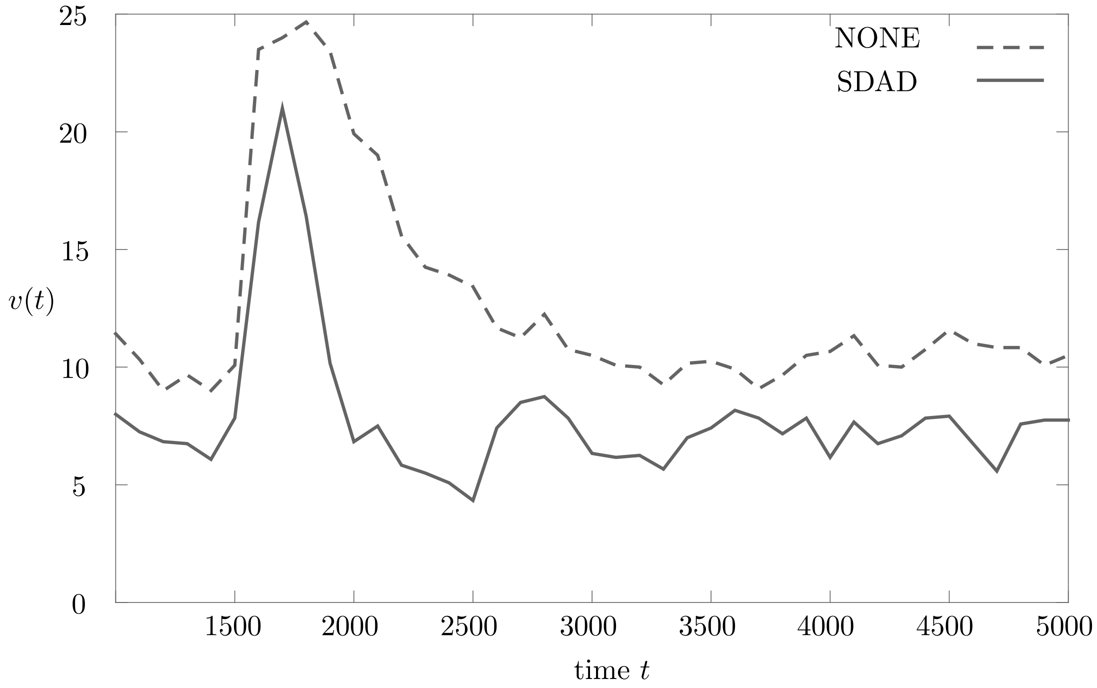

Figure 4: Deployed vehicles $v(t)$ from the own fleet ($\alpha = 3$)

increases up to $\phi(3) = 292.3$. If SDAD is applied then $\phi(3) = 226.6$ which means that SDAD supports the reduction of the number of waiting requests independently from the subcontracting tariffs.

The application of the SDAD strategy results in significantly improved transportation processes with respect to the punctuality. Since SDAD overrules the minimisation of costs, additional expenditures occur. The additional costs are shown in Tab. 5. In the NONE-experiment, a relative increase of costs of $c(3) = 42.8\%$ is observed. The cost increase observed in the SDAD-experiment is more severe than in the NONE experiment independently from the tariff level α. In conclusion, we state that additional costs are a significant drawback for the higher process reliability.

Table 4: Maximal number of waiting requests $\phi(\alpha)$ (numbers in pcs.)

	α					
	1	1.25	1.5	1.75	2	3
NONE	195.0	208.9	221.8	238.3	248.4	292.3
SDAD	191.9	199.1	210.9	217.0	218.8	226.6

Table 5: Increase $c(\alpha)$ of the overall costs (in percent)

	α					
	1	1.25	1.5	1.75	2	3
NONE	0.0	7.7%	13.3%	21.4%	27.0%	42.5%
SDAD	0.0	11.1%	16.9%	26.3%	32.2%	53.9%

Table 6: Cost increase

		α					
		1	1.25	1.5	1.75	2	3
NONE	$c^{INT}(\alpha)$	0.0	69.8%	119.3%	150.1%	168.3%	207.4%
	$c^{EXT}(\alpha)$	0.0	-39.8%	-66.7%	-76,7%	-81.6%	-91.3%
	$c^{PEN}(\alpha)$	0.0	80.9%	122.8%	168.5%	202.9%	356.0%
SDAD	$c^{INT}(\alpha)$	0.0	40.1%	71.6%	86.0%	114.5%	144.0%
	$c^{EXT}(\alpha)$	0.0	-1.8%	-7.4%	-0.2%	-3.9%	13.6%
	$c^{PEN}(\alpha)$	0.0	47.6%	82.8%	100.4%	120.2%	174.4%

In order to get a deeper understanding of the monetary impacts of the SDAD application, we analyse the amount of the three kinds of costs: travel costs of the own fleet (INT), subcontracting costs (EXT) and penalties for too-late arrivals (PEN). The development of these three cost drivers are presented in Tab. 6. We have observed different impacts of the tariff level increase. In the NONE-experiment the travel costs are triplicated (plus $c^{INT}(3) = 207.4\%$ monetary units) but in the SDAD-experiment the travel costs are raised only by 144.0%. The neglect of any model adaptation leads to a nearly complete disappearance of subcontracting costs as a response to a tariff increase ($c^{EXT}(3) = -91.3\%$) but in the SDAD-experiment the amount of externalisation costs remains nearly stable. A qualitatively as well as quantitatively different cost development is observed for the penalty costs. In the NONE-experiment, the compensation costs explode and a triplication of the tariff level α leads to an increase by the factor 4.5 (the amount is more than quadruplicated). Contrary to this observation, the triplication of the tariff level in the SDAD-experiment results in less than triplicated penalty costs (factor 2.74).

We finally analyse the contributions of the three cost drivers to the total costs (Tab. 7) and compare the percentage of travel costs, subcontracting costs and penalty costs. For quite fair states ($\alpha = 1$), subcontracting is the most important cost driver with $m^{EXT}(1) = 57.1\%$ observed in the NONE-experiment and $m^{EXT}(1) = 69.6\%$ of the

Table 7: Split of Costs

		α					
		1	1.25	1.5	1.75	2	3
NONE	$m^{INT}(\alpha)$	39.2%	61.8%	75.8%	80.7%	82.8%	84.5%
	$m^{EXT}(\alpha)$	57.1%	31.9%	16.8%	11.0%	8.3%	3.5%
	$m^{PEN}(\alpha)$	3.8%	6.3%	7.4%	8.3%	8.9%	12.0%
SDAD	$m^{INT}(\alpha)$	28.2%	35.6%	41.4%	41.5%	45.8%	44.7%
	$m^{EXT}(\alpha)$	69.6%	61.6%	55.2%	55.0%	50.6%	51.4%
	$m^{PEN}(\alpha)$	2.2%	2.9%	3.4%	3.4%	3.6%	3.9%

total costs observed in the SDAD-experiment. The second important driver is the travel costs with $m^{INT}(1) = 39.2\%$ of the total costs (NONE) and $m^{INT}(1) = 28.2\%$ (SDAD) respectively. Penalty costs do not contribute more than 3.8% to the total costs.

After the tariff lifting, the contributions of the cost drivers are different and differ with respect to the applied strategy. If the NONE-strategy is used then travel costs become the most important cost driver (84.2%), followed by the penalty costs (12.0%) and the subcontracting costs (3.5%). Here, the tariff lifting has lead to a nearly complete disregard of the subcontracting mode. The application of the SDAD-strategy results in quite different meanings: Both fulfilment modes share nearly the same part of more than 97% of the overall costs and penalties contribute only 3.9% to the overall costs. Here, the re-weighting of the two fulfilment modes has kept the subcontracting as a valuable alternative to the self-execution.

6 Conclusions

We have proposed and assessed an extension of the online decision making approach that allows an automatic situational adjustment of the formal problem representation. Here, we have adapted the objective function of the underlying optimisation model. This modification affects the search direction of the repeatedly used solving process in order to emphasise the search for reliable transport processes by overruling the cost criteria if necessary. The general applicability as well as the effects on reliability and costs have been quantified. Future research will address the comparison of the generic approaches to modify the search direction with the modification of the search space variation as well as the combination of the two approaches.

Acknowledgement. This research was supported by the German Research Foundation (DFG) as part of the Collaborative Research Centre 637 "Autonomous Cooperating Logistic Processes" (Subproject B7).

References

Bianchi, L., Birattari, M., Chirandini, M., Manfrin, M., Mastrolilli, M., Paquete, L., Rossi-Doria, O., SChiavinotto, Z., 2005. Hybrid metaheuristics for the vehicle scheduling problem with stochastic demands. Tech. Rep. IDSIA-0605, IDSIA.

Fleischmann, B., Gnutzmann, S., Sandvoß, E., 2004. Dynamic vehicle routing based on online traffic information. Transportation Science 38 (4), 420–433.

Gayalis, S., Tatsiopoulos, I., 2004. Design of an it-driven decision support system for vehicle routing and scheduling. European Journal of Operational Research 152 (2), 382–398.

Gendreau, M., Potvin, J.-Y., 1998. Dynamic vehicle routing and dispatching. In: Crainic, T., Laporte, G. (Eds.), Fleet Management and Logistics. Kluwer, pp. 115–126.

Ghiani, S., 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.

Gutenschwager, K., Böse, F., Voß, S., 2003. Effiziente Prozesse im kombinierten Verkehr – ein neuer Lösungsansatz zur Disposition von Portalkränen. Logistik Management 5 (1), 62–73.

Gutenschwager, K., Niklaus, C., Voß, S., 2004. Dispatching of an electric monorail system: Applying metaheuristics to an online pickup and delivery problem. Transportation Science 38 (4), 434–446.

Hiller, B., Krumke, S., Rambau, J., 2006. Reoptimization gaps versus model errors in online-dispatching of service units for adac. Discrete Applied Mathematics 154 (13), 1897–1907.

Jaillet, P., 1988. A priori solution of a traveling salesman problem in which a random subset of customers are visited. Operations Research 36 (6), 929–936.

Jensen, M., 2001. Robust and flexible scheduling with evolutionary computation. Ph.D. thesis, University of Aarhus.

Krumke, S., 2001. Online-Optimisation - Competitive Analysis and Beyond. Technical University of Berlin.

Lackner, A., 2004. Dynamische Tourenplanung mit ausgewählten Metaheuristiken. Cuvillier Verlag, Göttingen.

Mitrović-Minić, S., Krishnamurti, R., Laporte, G., 2004. Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. Transportation Research Part B 38, 635–655.

Pankratz, G., 2002. Speditionelle Transportdisposition. Deutscher Universitäts-Verlag, Wiesbaden.

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

Savelsbergh, M., Sol, M., 1998. Drive: Dynamic routing of independent vehicles. Operations Research 46, 474–490.

Schönberger, J., 2005. Operational Freight Carrier Planning. GOR Publications. Springer, Berlin Heidelberg.

Schönberger, J., Kopfer, H., 2007. On decision model adaptation in online optimization of a transport system. In: Günther, H.-O., Mattfeld, D., Suhl, L. (Eds.), Management logistischer Netzwerke. Springer, Berlin Heidelberg, pp. 361–381.

Séguin, R., Potvin, J.-Y., Gendreau, M., Crainic, T., Marcotte, P., 1997. Real-time decision problems: an operational research perspective. Journal of the Operational Research Society 48 (2), 162–174.

Solomon, M., 1987. Algorithms for the vehicle routing and scheduling problem with time window constraints. Operations Research 35 (2), 254–265.