Logist. Res. (2011) 3:121–132
DOI 10.1007/s12159-011-0055-8

REVIEW ARTICLE

Multicriteria decentralized decision making in logistic chains: a dynamic programming approach for collaborative forwarding of air cargo freight

Martin Berger · Michael Schröder

Received: 30 September 2010 / Accepted: 30 March 2011 / Published online: 16 April 2011
© Springer-Verlag 2011

Abstract Forwarding air freight cargo globally from the shipper's door to the door of the consignee is a complex logistic process and involves freight handling by several collaborating logistics companies. The door to door process is currently standardized by the International Air Transport Association with the industry initiative Cargo 2000. Many individuals with their own perspectives along the logistic chain decide how freight is transported and handled but have only limited insights how their decisions influence the follow-up decisions and the final on time delivery. A central planning authority can not be realized due to the heterogeneity of decision makers, the individual interests of the logistics companies, and their global operations. We argue that it is possible to overcome the deficient situation of complex transportation chains like in the air cargo industry by using optimization methods for decentralized decision making. This paper proposes a dynamic programming approach, which enables the decision makers in the decentralized situation to align their decisions better with the decisions of the involved partners. The approach guarantees that sensitive information of the logistics companies is kept local and only the most necessary information is shared along the logistic chain for a better planning. The transportation is planned with regard to multiple criteria, like the expected transportation costs and the probability to deliver the freight on time. We further show that our decentralized and multicriteria approach leads to better results compared to a local strategy that only exploits each decision maker's own perspective. Our approach is decentralized by nature and needs lean information exchange. Furthermore, it is as strong as a centralized approach that gathers all distributed information but that authorizes the logistic service providers to decide individually.

Keywords Decentralized decision making · Multicriteria optimization · Logistic chains · Dynamic programming · Air cargo freight

This research is developed within the project ADiWa funded by means of the German Federal Ministry of Education and Research under the promotional reference 01IA08006. The authors take the responsibility for the contents.

M. Berger (✉) · M. Schröder
Fraunhofer Institute for Industrial Mathematics (ITWM),
Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany
e-mail: martin.berger@itwm.fraunhofer.de

1 Introduction

1.1 Forwarding air cargo freight

To put it simply, the business of the air cargo industry is to forward air freight from the shipper to the consignee. However, in the logistics sense, forwarding air cargo is a highly dynamic process and complex due to many factors: the global transnational destinations; time shifts; language barriers; many collaborating logistics companies like airlines, freight forwarders, ground handling agents, and trucking companies.

The International Air Transport Association is currently forcing an industry initiative called Cargo 2000, which aims at quality and efficiency improving standards for the forwarding process of air freight. In the following, we call this process the door to door (D2D) process because it starts at the door of the shipper and ends at the door of the consignee. It is operated via a freight carrier network [19, pp. 4–9] that is defined by the involved logistics

Fig. 1 Door to door process with the following main locations from *left* to *right*: shipper; export warehouse; export hub; export ground handling agent; import ground handling agent; import hub; import warehouse; consignee

companies. The main locations of a D2D process are illustrated in Fig. 1.

The D2D process is executed by many decision makers of different logistics companies, which have their own perspective of the overall logistic chain from door to door. Each individual controls one chain link and has to decide how air freight is handled and transported in his authority. Collaboration enables them to operate on a global market and contracted service agreements guarantee that the business is profitable for all of them.

However, besides the agreements, most of the decision makers have only limited insights how their decisions influence the decisions of follow-up logistic partners. How their decisions influence the overall performance of the freight forwarding, whether goods are delivered on time and as initially requested, is hardly to predict. Nowadays there is no information technology (IT) infrastructure that entirely supports planning and execution of air freight forwarding from door to door. Consequently, the decision makers coordinate their local planning only loosely via telephone, fax, or email.

1.2 Reasons for a decentral planning concept

Decision support for planning transportation and handling options has great potential to improve air freight forwarding in the D2D process. However, we claim that a central planning authority for the air cargo D2D process can not be realized due to several reasons:

1. The planning has to be done across organizational boundaries. Often the collaborating companies along the D2D chain have only a loose relationship, sometimes established ad hoc for a single transport. Their IT systems span a wide spectrum and are typically not integrated. Only lean interfaces providing a small amount of standardized information seem realistic.
2. It is hard to imagine how a central planning authority for the D2D operations could be established. It would have to be international by nature; it has to integrate world-wide operating logistics companies and regional forwarders, together with a multitude of local handling agents in many countries. It would have to access data relevant for planning from all these companies. Furthermore, it must be authorized to impose an overall plan with executable operations on all companies.
3. The planning process has to address the interest of the collaborating companies to keep a large part of planning relevant data internal, most notably, the cost structures. Even if some information, e.g., for flights, are available in public, most collaborating companies want to maintain sensitive data, e.g., costs, transportation times, and the reliability of their fleet, only in their private IT environment. A central planning authority that must have access to such internal data during plan computation would not be accepted.

From these considerations, it is clear that only a decentralized planning concept can be realized for the D2D process. Our approach advances from a completely independent planning to a loosely coupled coordinated planning and resembles other concepts of collaboration in supply chains, like collaborative planning, forecasting, and replenishment (CPFR).

1.3 Planning problems in D2D forwarding

When the customer requests forwarding of a single air cargo freight, the decision maker responsible for the customer contact promises that the freight is delivered at a certain time. The milestone "proof of delivery" (POD) of Cargo 2000 results from the pick-up time of the freight at the shipper and the actual promised time that is needed to deliver the freight to the consignee. The POD milestone marks a delivery deadline to which we refer to as POD time in the following.

Whether the decision maker can promise the requested POD time results from an initial planning problem. Nowadays, this problem is independently solved for each customer request by using rough transportation times for the freight carrier network. The times are unrealistically assumed to be deterministic. The POD time results from accumulating transportation times along a D2D chain selected by the decision maker. Decision makers of the D2D process also face an adaptive replanning problem whenever delays occur. For the currently authorized decision maker, it is hard to decide under time pressure whether a delay really endangers the POD time and how the schedule has to be replanned. Often decisions are made based on experience only or by suboptimal coordination with the follow-up decision makers. In this paper, we address both the initial and the adaptive planning problem of the D2D process and propose a decentralized approach that can be used to solve both planning problems.

1.4 Literature review

Transportation planning problems for freight forwarding are intensively studied in the literature. We review freight transportation in general and discuss optimization problems and solution approaches that are related to our problem. Finally, we evaluate research that regards the differences of central and decentral organization of transportation problems.

A good overview of transportation problems is given in [8]. The authors of [11] discuss planning models for long-haul operations of post and express freight. However, the proposed central decision support is designed for a single logistics company that is responsible for the entire forwarding process. In [14], a planning approach for landside forwarding less-than-truckload shipments is proposed that integrates hired subcontractors besides the trucking company's fleet. Research for air cargo has mainly focused on air cargo assignment to capacities of the airlines. In [3], a mathematical programming approach to revenue management in cargo airlines is studied. Optimization models for air container and cargo loading problems in freight forwarding are proposed in [22]. Both a deterministic and a stochastic problem under uncertainty are solved by means of mixed integer programming, stochastic and robust optimization. However, the paper only discusses the renting and loading of the freight of the airlines and ignores the forwarding from the door of the shipper to the carrier as well as the forwarding after the flight to the door of the consignee. In [5], a multiobjective dynamic programming approach for intermodal transport planning is proposed but the approach disregards uncertainty and is not proposed for a decentralized setup.

Planning the transportation of air cargo freight is strongly related to multicriteria shortest path problems (MSPP). MSPPs are solved for example with labeling algorithms and dynamic programming [9, 13, 18] or with Markov decision processes [21]. A comparison for some of the solution methods is given in [17, 20]. The authors of [7] propose an interactive approach for bicriteria shortest path problems that uses a k-shortest path procedure discussed in [2]. Most methods for MSPP are designed for central planning and it is often not straight forward to apply them in a decentralized setup. In [1], a decentralized algorithm is proposed that finds shortest paths in dynamic networks. However, this approach does not address stochastic edge lengths and time-dependent availability of edges to model delays and missings of follow-up transports. Stochastic shortest path problems are also widely studied [4, 15]. However, research on stochastic problems that also considers multiple criteria is rare [16].

In [12], a paradigm change in logistics from central control and planning toward decentralized and autonomous perspectives and its implications for decision making are studied. In [10], different decision making structures for logistic systems are discussed: hierarchy, heterarchy, responsible autonomy, and anarchy. In order to meet the requirements of the D2D process, a decentralized approach for decision support must be addressed that delegates control in heterarchic and hierarchic structures, i.e., heterarchic local planning and hierarchic global planning.

To our best knowledge, no existing approach in the literature simultaneously addresses multicriteria decentralized transportation planning under uncertainty with fixed timetables for the start times of the transports, a setup we face in forwarding air cargo freight.

1.5 Contribution of this work

We argue that it is possible to overcome the deficient situation of complex transportation chains like in the air cargo industry by using optimization methods for decentralized decision making. We propose a dynamic programming approach that enables the decision makers in the decentralized situation to make decisions better aligned with decisions from others. The approach guarantees that sensitive information of the logistics companies are kept local and only the most necessary information is shared along the logistic chain for a better planning. The transportation is planned with regard to uncertainty and multiple criteria, like the expected transportation costs and the probability to deliver the freight in the desired POD. Our approach solves both the initial and adaptive planning problem of the D2D forwarding process and provides the decision makers an evaluation of their transport options with respect to the optimization criteria.

We further show that our decentralized and multicriteria approach leads to better results compared to a local strategy that only exploits each decision maker's own perspective. We prove that our algorithm has a polynomial runtime, show that it is decentralized by nature and needs lean information exchange without sharing sensitive data. Through its concept, it is as strong as a centralized approach that gathers all distributed information but that still authorizes the logistic service providers to decide individually. Hence, it is not necessary to maintain distributed information at a central instance.

1.6 Outline of the paper

In Sect. 2, we introduce the planning model our algorithms work on. In Sect. 3, we introduce the multicriteria dynamic program that is solved in the decentralized D2D setup. We compare this algorithm to a local strategy heuristic.

We evaluate the algorithms in Sect. 4. In the Sect. 4.1, we present a theoretical analysis. Section 4.2 shows several empirical studies based on simulations of the D2D process where transportation is initially planned and in case of delays adaptively replanned. We conclude the paper in Sect. 5.

2 Planning model

Our decentralized decision support is designed for a decision maker who has to plan the forwarding of a single air freight consignment. It supports him in two situations: when he has to assign an initial forwarding plan for the air freight; and when he has to reschedule the forwarding due to severe disruptions during execution of the D2D process. Our methods work for both the initial planning and the replanning. Without loss of generality, we mainly focus on the initial planning situation because the replanning can be viewed as initial planning for the decision maker who is currently in control of the forwarding. Interdependencies between cargo are disregarded in the global planning but we assume that they are locally regarded in the transport planning of each decision maker. We keep the global planning problem simple in this way but locally regard the effects of interdependencies.

We consider a planning situation in which the air freight must be picked up at the shipper at time a_0 and whose POD is at time t_{POD}. Our approach relies on a stochastic model of the planning situation that regards delays in transportation. For simplicity's sake, we consider the transport steps of the forwarding as being binomial distributed: a transport is on time with a probability $p \in [0, 1]$ and delayed with probability $1 - p$. Our approach can straightforwardly be extended to more detailed probability distributions without increasing the worst-case runtime complexity.

We assume an acyclic transport network $G(L, E)$ to send air freight from a shipper to a consignee. The nodes L of the network are possible locations of the D2D chain. The edges E represent transport relations that connect the locations and that have several transport options attached to forward the air freight. Without loss of generality, we assume that there is at least one path from the shipper to every location in the transport network and that there is at least one path from every location to the consignee. Hence, we assume that $G(L, E)$ is a weakly connected graph. These assumptions are no restrictions since locations that are not on a path from the shipper and that are not on a path to the consignee cannot be used within the D2D process chain. An example for a transport network topology is given in Fig. 2 to which Table 1 associates possible D2D locations.

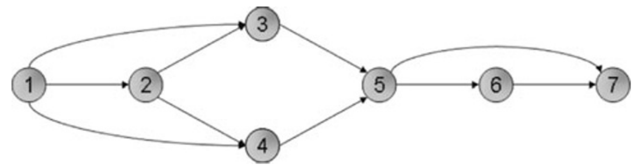

Fig. 2 Example for a transport network topology

Table 1 Possible D2D locations for the nodes of the transport network of Fig. 2

Node	D2D location
1	Shipper
2	Export warehouse
3	Export hub 1
4	Export hub 2
5	Import hub
6	Import warehouse
7	Consignee

In such a given transport network with n locations, the sink location L_n represents the consignee and the source location L_1 represents the shipper. We formalize the connections between the locations as follows:[1] a function $e^+ : L \rightarrow 2^L$ returns all direct predecessor locations that forward to a location and a function $e^- : L \rightarrow 2^L$ gives all direct successor locations of a location. When the context is clear, we simply use the location indices in these functions, i.e., we use functions $e^+, e^- : \{1, \ldots, n\} \rightarrow 2^{\{1,\ldots,n\}}$.

Furthermore, we have transport options for all transport relations in E: The kth transport option T_{ij}^k from location L_i to location L_j is described by a starting time s_{ij}^k, a regular arrival time a_{ij}^k and a delayed arrival time l_{ij}^k at the next location L_j. The binomial distributed lateness probability is expressed as follows: The on time arrival time a_{ij}^k is realized with a probability p_{ij}^k whereas a delayed arrival time l_{ij}^k is realized with probability $1 - p_{ij}^k$. The costs c_{ij}^k for the transport T_{ij}^k are known for each option and are assumed to be independent of delays.

Our decentralized decision support provides the current decision maker in the D2D process an evaluation of his transport options that reflects the influence of his decisions on the follow-up forwarding operations. The decision maker at location L_i of the D2D chain who receives an air freight item at time t, gets an evaluation of his transport options with respect to the following two criteria:

1. The criterion $C(i, t)$ describes the expected costs for the remaining transport chain from location L_i to final destination L_n,
2. The criterion $P(i, t)$ describes the probability to achieve the POD at time t_{POD} by taking possible delays into account.

[1] The notation 2^L means the power set of the set L.

Table 2 Summary of the notation used in this paper

a_0	$\in \mathbb{R}_0^+$	Time of pick-up at L_1
t_{POD}	$\in \mathbb{R}_0^+$	POD time at the consignee
n	$\in \mathbb{N} \setminus \{0\}$	Number of locations of the transport network
L_i		ith location of the transport network, $i \in \{1,\ldots,n\}$
L_1		Location of the shipper
L_n		Location of the consignee
m_{ij}	$\in \mathbb{N} \setminus \{0\}$	Number of transport options for forwarding from location L_i to location L_j
e^+:	$L \to 2^L$	Function $e^+(L_i)$ returns all direct predecessor locations of location L_i
e^-:	$L \to 2^L$	Function $e^-(L_i)$ returns all direct successor locations of location L_i
T_{ij}^k		kth transport from L_i to L_j, $k \in \{1,\ldots,m_{ij}\}, i \in \{1,\ldots,n-1\}, j \in \{2,\ldots,n\}$
s_{ij}^k	$\in \mathbb{R}_0^+$	Start time of T_{ij}^k at L_i
a_{ij}^k	$\in \mathbb{R}_0^+$	On time arrival time of T_{ij}^k at L_j
l_{ij}^k	$\in \mathbb{R}_0^+$	Delayed arrival time of T_{ij}^k at L_j
p_{ij}^k	$\in [0,1]$	Probability of on time arrival of T_{ij}^k
c_{ij}^k	$\in \mathbb{R}_0^+$	Cost of T_{ij}^k
$C(i, t)$		Expected costs for transportation from L_i to L_n starting at time t
$P(i, t)$		Probability for delivery at L_n before time t_{POD} by starting from L_i at time t
α_i	$\in [0,1]$	Business policy
		$\alpha_i = 1$: maximize POD probability
		$\alpha_i = 0$: minimize expected costs

The decision maker of each location L_i follows a business policy α_i that represents a weight between the criteria for the expected costs and the POD probability. A value of one yields a business policy in which the decision maker only wants to maximize the probability of a POD whereas a value of zero represents his intention to minimize expected costs. The model could be easily extended to more than these two criteria.

The air freight forwarding is successively planned with respect to the expected costs $C(i, t)$ and the overall POD probability $P(i, t)$ by solving a bicriteria stochastic optimization problem for the current location L_i at time t with respect to the remaining logistic chain. The notation described above is summarized in Table 2.

3 Methods

In this section, we introduce two methods to solve our initial and adaptive planning problems for the D2D process. First, we propose a dynamic programming approach and then we describe a local strategy heuristic.

3.1 Decentralized dynamic programming

Dynamic programming is a method for efficiently solving a broad range of search and optimization problems that exhibit the following two characteristics:

1. The problem can be broken down into easy subproblems, which are reused multiple times.
2. The global optimal solution can be constructed from locally optimal solutions to the subproblems.

These characteristics enable us to approach the forwarding problem in a decentralized way. The local planning subproblem is solved at each location and the global solution is constructed by combining the local solutions sharing only the expected costs and the POD probability between the locations.

A dynamic program can be solved by recursion. In the following, we describe the main steps of the recursive algorithm and use an additional notation for the set of possible arrival times at a location L_j:

$$A_j := \left\{ a_{ij}^k \middle| \ \forall L_i \in e^+(L_j), \quad k = 1,\ldots,m_{ij} \right\} \cup \left\{ l_{ij}^k \middle| \ \forall L_i \in e^+(L_j), \quad k = 1,\ldots,m_{ij} \right\}, \quad j = 2,\ldots,n,$$

$$A_1 := \{a_0\}.$$

3.1.1 Initialization

Our algorithm initially calculates the expected costs $C(n, t)$ and the POD probability $P(n, t)$ of the consignee location L_n for all $t \in A_n$ as follows:

$$C(n,t) := 0,$$

$$P(n,t) := \begin{cases} 1, & \text{if } t \le t_{POD}, \\ 0, & \text{else.} \end{cases}$$

3.1.2 Recursion

We assume that the expected costs $C(j, t)$ and the POD probability $P(j, t)$ were calculated in the previous recursion steps for all times $t \in A_j$ and for all successor locations $L_j \in e^-(L_i)$ of location L_i. Then we can calculate $C(i, t)$ and $P(i, t)$ for all times $t \in A_i$ as follows:

- For all $k \in \{1,\ldots,m_{ij}\}$ where the start times s_{ij}^k are earlier then the current time t, i.e., $s_{ij}^k < t$, we set the expected costs and the POD probability as follows:

  $$C(i,t) = 0, \quad P(i,t) = 0.$$

- Let $K(i,j,t) := \{k | k \in \{1,\ldots,m_{ij}\} \wedge s_{ij}^k \geq t\}$ be indices of the feasible transport options from L_i to L_j at time t. Furthermore, let $s_{ij} := \min_{k\in\{1,\ldots,m_{ij}\}} s_{ij}^k$ be the earliest start time of a transport option from L_i to L_j. For the indices $k \in K(i,j,t)$ we calculate the expected costs and the POD probability as follows:

  $$\widetilde{C}(i,j,t,k) := c_{ij}^k + p_{ij}^k C\left(j, a_{ij}^k\right) + \left(1 - p_{ij}^k\right) C\left(j, l_{ij}^k\right),$$
  $$\widetilde{P}(i,j,t,k) := p_{ij}^k P\left(j, a_{ij}^k\right) + \left(1 - p_{ij}^k\right) P\left(j, l_{ij}^k\right),$$
  $$\overline{C} := \sum_{j\in e^-(i)} \sum_{k\in K(i,j,s_{ij})} \widetilde{C}(i,j,t,k),$$
  $$\overline{P} := \sum_{j\in e^-(i)} \sum_{k\in K(i,j,s_{ij})} \widetilde{P}(i,j,t,k).$$

  We define the indices $j^* \in e^-(i)$ and $k^* \in K(i,j^*,t)$ as follows:

  $$(j^*,k^*) := \operatorname*{arg\,min}_{j\in e^-(i),\, k\in K(i,j,t)} (1-\alpha_i)\frac{\widetilde{C}(i,j,t,k)}{\overline{C}} - \alpha_i \frac{\widetilde{P}(i,j,t,k)}{\overline{P}}.$$

  The index j^* yields the locally optimal transport connection and the index k^* the locally optimal transport option $T_{ij^*}^{k^*}$ from L_i to L_{j^*} according to the business policy α_i. Finally, the expected costs and the POD probability for L_i at the time t are as follows:

  $$C(i,t) := \widetilde{C}(i,j^*,t,k^*), \quad P(i,t) := \widetilde{P}(i,j^*,t,k^*).$$

3.1.3 Termination

The recursion terminates when the expected costs $C(1,a_0)$ and the POD probability $P(1,a_0)$ for the first location L_1 are calculated.

3.1.4 Algorithm

The dynamic program is summarized in Algorithm 1 from a centralistic point of view.

The algorithm works in an analogous way in the decentralized setup, where the business policies and the data of the transport options are not shared between the involved locations. The locations are triggered in the same order as in Algorithm 1 but the triggering is obtained implicitly and not controlled with a central queue. The decentralized algorithm at each location buffers the planning results of the follow-up locations. When all successors have sent their results, which can be deduced with help of lines 3–5 and lines 17–21 of Algorithm 1, it calculates $C(i, t)$ and $P(i, t)$ (line 16 of Algorithm 1) with the locally known business policy α_i. The calculated $C(i, t)$ and $P(i, t)$ for all relevant times t are sent to the decentralized algorithms of the predecessor locations $e^+(L_i)$. The last planning step is done at the first location L_1, which is currently in execution and can evaluate different expected costs $C(1,a_0)$ and corresponding POD probabilities $P(1,a_0)$ by varying its own business policy α_1.

Algorithm 1 Pseudocode of the multicriteria dynamic program

Require: business policies α_i of locations $L_i \in L$
1: initialize A_i for all locations $L_i \in L$
2: initialize $C(n, t)$ and $P(n, t)$ as described in Sect. 3.1.1
3: **for all** $L_i \in L$ **do**
4: nrSucc$[L_i] \leftarrow |e^-(L_i)|$
5: **end for**
6: // initialize queue Q of processable locations (all successors processed)
7: **for all** $L_i \in e^+(L_n)$ **do**
8: **if** $|e^-(L_i)| = 1$ **then**
9: Q.enque(L_i)
10: **else**
11: nrSucc$[L_i] \leftarrow$ nrSucc$[L_i]$-1
12: **end if**
13: **end for**
14: **while** Q is not empty **do**
15: $L_i \leftarrow Q$.deque()
16: calculate $C(i, t)$ and $P(i, t)$ for all $t \in A_i$ (recursion step described in Sect. 3.1.2)
17: **for all** $L_j \in e^+(L_i)$ **do**
18: nrSucc$[L_j] \leftarrow$ nrSucc$[L_j]$-1
19: **if** nrSucc$[L_j] = 0$ **then**
20: Q.enque(L_j)
21: **end if**
22: **end for**
23: **end while**
24: **return** $C(1, a_0)$, $P(1, a_0)$, j^* and k^* for all L_i and all $t \in A_i$

3.2 Local strategy heuristic

In this subsection, we propose a local strategy heuristic that only regards the point of view of the current decision maker with his own business policy.

When the air freight item is at location L_i at time t, the local strategy heuristic evaluates all transport options T_{ij}^k of L_i whose start time is no earlier than the current time, i.e., $s_{ij}^k \geq t$. Let $K(i,j,t) := \left\{k | k \in \{1,\ldots,m_{ij}\} \wedge s_{ij}^k \geq t\right\}$ be again the indices of the feasible transport options from L_i to L_j at time t and $s_{ij} := \min_{k \in \{1,\ldots,m_{ij}\}} s_{ij}^k$ be the earliest start time of a transport option from L_i to L_j.

The evaluation includes the expected arrival time of the feasible transport options as well as the costs. The expected arrival time $E_k(j)$ of a transport option T_{ij}^k at the next location L_j is calculated as follows:

$$E_k(j) := p_{ij}^k a_{ij}^k + \left(1 - p_{ij}^k\right) l_{ij}^k$$

The costs c_{ij}^k are constant independent of the on time probability.

We normalize the expected arrival time and the costs of a transport option $T_{ij}^{\hat{k}}$ where $\hat{k} \in K(i,j,t)$ as follows:

$$f(E_{\hat{k}}(j)) := \frac{E_{\hat{k}}(j) - \min_{j \in e^-(i),\, k \in K(i,j,s_{ij})} E_k(j)}{\max_{j \in e^-(i),\, k \in K(i,j,s_{ij})} E_k(j) - \min_{j \in e^-(i),\, k \in K(i,j,s_{ij})} E_k(j)} \tag{1}$$

$$f\left(c_{ij}^{\hat{k}}\right) := \frac{c_{ij}^{\hat{k}} - \min_{j \in e^-(i),\, k \in K(i,j,s_{ij})} c_{ij}^k}{\max_{j \in e^-(i),\, k \in K(i,j,s_{ij})} c_{ij}^k - \min_{j \in e^-(i),\, k \in K(i,j,s_{ij})} c_{ij}^k} \tag{2}$$

We define the indices $j^* \in e^-(i)$ and $k^* \in K(i,j^*,t)$ as follows:

$$(j^*, k^*) := \underset{j \in e^-(i),\, k \in K(i,j,t)}{\arg\min} (1 - \alpha_i) f\left(c_{ij}^k\right) + \alpha_i f(E_k(j)).$$

The index j^* gives us the transport connection for the index k^* of the best transport option T_{ij^*} from L_i to L_j according to the business policy α_i of the decision maker and his local view on the transport network.

4 Evaluation

Now, we evaluate the methods proposed in the previous section with respect to theoretical and empirical aspects. First we analyze the algorithms theoretically and then we present studies evaluating the algorithms empirically.

4.1 Theoretical analysis

First, we show that the dynamic program is suitable for decentralized planning and decision making. Next, we prove that our multicriteria dynamic program has a polynomial worst-case runtime complexity. Finally, we observe differences between the local strategy and our dynamic program.

4.1.1 *Lean information exchange for decentralized planning*

Our dynamic programming approach exploits information of the entire transport network but one important property of it is that it must not exchange all local information like

- the business policies α_j,
- the transport data T_{jx}^k, a_{jx}^k, l_{jx}^k, p_{jx}^k,
- and their cost structure c_{jx}^k.

The only information that needs to be propagated backwards in the network are possible times to process the freight, i.e.,

$$t \in \left\{s_{jx}^k \,|\, k = 1,\ldots,m_{jx}, \quad \forall L_x \in e^+(L_j)\right\},$$

the corresponding expected costs $C(j, t)$ and the POD probability $P(j, t)$ for further forwarding it to the consignee. Each location inserts its local information in the planning and implicitly propagates it backwards in the network. Figure 3 illustrates what data have to be exchanged between locations and what data are sealed locally.

The risk that sensitive information can be reengineered by means of executing the algorithm multiple times with slightly changed input parameters is low because the exchanged information is aggregated with information of several follow-up logistic service providers. It is hard to deduce individual data of decision makers just with the help of the aggregated criteria. Only the direct predecessor of the last decision maker can potentially gain insights if he knows that he directly precedes the last decision maker. However, such an attack is unlikely, rather artificial and can be easily prevented by assuring decision makers do not know whether they are direct predecessors of the last decision maker.

Our dynamic program could also be executed by a central authority who knows all information. However, we state that this algorithm also works perfectly in the described decentralized setup without the need to maintain all relevant information from a central instance.

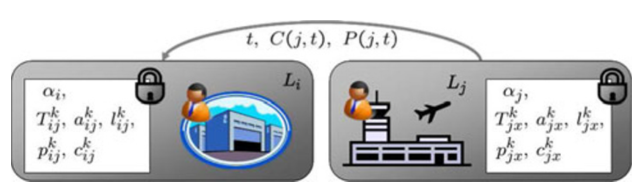

Fig. 3 Illustration that shows which data need to be exchange and what data are sealed locally by our planning approach

4.1.2 Worst-case runtime complexity of our dynamic program

Theorem 1 *The worst-case runtime complexity of Algorithm 1 is polynomial in the number of transport connections and number of transport options, i.e., it runs in*

$$O\left(\sum_{L_i \in L \setminus \{L_n\}} \sum_{j \in e^-(i)} m_{ij}\right).$$

Proof We first show that each location $L_i \in L \setminus \{L_n\}$ is processed exactly once. Our algorithm executes a breadth-first search for which each reachable node is enqueued and dequed only once [6, p. 531]. We assumed a directed acyclic transportation network where at least one path exists from the shipper node to all other nodes and where at least one path exists from all nodes to the consignee node. Therefore, each location node is processed exactly once.

For each dequed location $L_i, i = 2, \ldots, n$ we have to calculate the best transport option $T_{ij^*}^{k^*}$. This is achieved by investigating all relations to the follow-up locations $L_j \in e^-(L_i)$ and the associated transport options T_{ij}^k. For the relation between L_i and L_j we have m_{ij} transport options. Thus, we finally yield the runtime complexity $O\left(\sum_{L_i \in L \setminus \{L_n\}} \sum_{j \in e^-(i)} m_{ij}\right)$. □

4.1.3 Comparison of algorithms

Now, we compare the two algorithms of the previous Sect. 3. The advantage of the local strategy heuristic is that no information exchange between the locations is necessary. However, the lack of information can lead to decisions that are locally optimal but which are suboptimal in the overall process chain. The following observation gives an example for a simple chain as transport network.

Observation 2 *In the following we observe a simple example where the local strategy heuristic that minimizes the expected arrival time of air fright performs worse compared to the dynamic programming approach. The simple example consists of three locations of a chain as transport network. The decision makers of the chain all try to maximize the POD probability, i.e.*

$$\alpha_i = 1, \quad i = 1, 2, 3.$$

The freight can be picked up at $a_0 = 7$ am and its POD is at $t_{\mathrm{POD}} = 1$ pm. The transport options of the transport chain are shown in Table 3. We consider the planning problem of the decision maker for the first transportation step.

The local strategy algorithm uses the expected arrival times

Table 3 Transport options of the simple transport chain

T_{ij}^k	s_{ij}^k	a_{ij}^k	l_{ij}^k	p_{ij}^k
T_{12}^1	7	8	20	0.9
T_{12}^2	7	8	12	0.5
T_{23}^1	8	13	14	0.9
T_{23}^2	12	13	14	0.8
T_{23}^3	20	21	22	0.7

$$E_1(2) = 0.9 \cdot 8 + 0.1 \cdot 20 = 9.2$$
$$E_2(2) = 0.5 \cdot 8 + 0.5 \cdot 12 = 10$$

and proposes to use the transport option ${T^1}_{12}$.

The dynamic programming approach propagates the POD probabilities for all arrival times from the last location to the first location:

$$\begin{aligned}\widetilde{P}(2,3,8,1) &= p_{23}^1 P\left(3, a_{23}^1\right) + \left(1 - p_{23}^1\right) P\left(3, l_{23}^1\right)\\ &= 0.9 \cdot 1 + 0.1 \cdot 0 = 0.9\\ \widetilde{P}(2,3,8,2) &= p_{23}^2 P\left(3, a_{23}^2\right) + \left(1 - p_{23}^2\right) P\left(3, l_{23}^2\right)\\ &= 0.8 \cdot 1 + 0.2 \cdot 0 = 0.8\\ \widetilde{P}(2,3,8,3) &= p_{23}^3 P\left(3, a_{23}^3\right) + \left(1 - p_{23}^3\right) P\left(3, l_{23}^3\right)\\ &= 0.7 \cdot 0 + 0.3 \cdot 0 = 0.0\end{aligned}$$

$$\begin{aligned}P(2,8) &= \max\left(\widetilde{P}(2,3,8,1), \widetilde{P}(2,3,8,2), \widetilde{P}(2,3,8,3)\right)\\ &= 0.9\\ P(2,12) &= \max\left(\widetilde{P}(2,3,12,1), \widetilde{P}(2,3,12,2), \widetilde{P}(2,3,12,3)\right)\\ &= 0.8\end{aligned}$$

$$\widetilde{P}(1,2,7,1) = 0.9 \cdot 0.9 + 0.1 \cdot 0.0 = 0.81$$
$$\widetilde{P}(1,2,7,2) = 0.5 \cdot 0.9 + 0.5 \cdot 0.8 = 0.85$$

The transport option ${T^2}_{12}$ finally has the higher POD probability of the two options of the decision maker in the first location. Therefore, the dynamic programming approach proposes to use ${T^2}_{12}$.

4.2 Empirical studies

In this subsection, we evaluate the local strategy heuristic and the dynamic programming algorithm by empirical studies. First, we compare the two algorithms and then we present an example of how the dynamic program can be applied in planning situations.

4.2.1 Comparison of algorithms

We first simulate an example of a D2D process on a transport network as it is illustrated in Fig. 2 and transport options given in Table 4. The transport network is discussed with experts of the air cargo industry and represents

Table 4 Transport options for the transport network of Fig. 2

T_{ij}^k	s_{ij}^k	a_{ij}^k	l_{ij}^k	p_{ij}^k	c_{ij}^k
T_{12}^1	6	7	8	0.95	100
T_{12}^2	6	7.5	8.5	0.98	90
T_{12}^3	7	8.5	9	0.90	80
T_{13}^1	7	10	11.5	0.92	250
T_{14}^1	7	10.5	11.25	0.99	270
T_{23}^1	7	10	11	0.95	100
T_{23}^2	7.5	11	12	0.98	80
T_{23}^3	9	12	12.5	0.90	70
T_{24}^1	7	10	11	0.95	100
T_{24}^2	7.5	11	12	0.96	65
T_{35}^1	10	14	15	0.95	210
T_{35}^2	11	14	15	0.90	200
T_{35}^3	13	15	16	0.50	100
T_{45}^1	10	14	15	0.95	210
T_{45}^2	11	14	15	0.90	200
T_{45}^3	12	15	16	0.92	200
T_{56}^1	14	15	16	0.90	200
T_{56}^2	15	16	17	0.70	100
T_{56}^3	19	20	21	0.45	90
T_{57}^1	15	16.5	17.5	0.99	250
T_{67}^1	15	16	17	0.90	200
T_{67}^2	16	17	17.5	0.70	100
T_{67}^3	21	21.5	22	0.45	90

a small but realistic and typical network structure: for example the nodes 3 and 4 represent alternative export hubs; the transport connections from the shipper node 1 directly to the export hubs 3 and 4 and the connection from the import hub node 5 to the consignee node 7 represent transport options that skip the warehouses 2 and 6. The data of the transport options are fictive but where designed to model a realistic situation. In the given example, an air freight item should be picked up at the shipper location L_1 at $a_0 = 6$ am and should be delivered at the consignee location L_7 at $t_{\mathrm{POD}} = 5$ pm.

We simulate the D2D process via the transport network in two scenarios:

1. We choose 21 equal business policies for the locations of the transport network, i.e.,

   $$\alpha(q) = \frac{q}{20}, \ \alpha_i = \alpha(q), \quad \forall L_i \in L, \ q = 0, 1, \ldots, 20.$$

   Equal business policies can be assumed for D2D processes where all decision makers have equal preferences regarding their decision criteria.
2. We uniformly choose 20 independent business policies for all locations, i.e.,

   $$\alpha_i(q) = \mathrm{rand}(0, 1), \quad \forall L_i \in L, \ q = 0, 1, \ldots, 19.$$

   Independent business policies can be assumed when each decision maker has different preferences, e.g., when they belong to autonomous logistics companies.

For all business policy setups with both algorithms, the forwarding of the air freight is simulated 1,000 times by uniformly choosing on time arrivals for the transport options.

The mean μ and standard deviation σ for the realized overall costs and the POD are evaluated for all simulated forwardings. The rounded results for the equal business policies are given in Table 5. Table 6 shows the results for uniformly chosen independent business policies.

For evaluating the numerical results, we utilize a dominance relation between the outcomes. Whenever Algorithm A dominates the mean results of Algorithm B (μ-dominates), i.e.,

$$(\mu_A(\mathrm{cost}) \leq \mu_B(\mathrm{cost}) \wedge \mu_A(\mathrm{POD}) > \mu_B(\mathrm{POD})) \\ \vee (\mu_A(\mathrm{cost}) < \mu_B(\mathrm{cost}) \wedge \mu_A(\mathrm{POD}) \geq \mu_B(\mathrm{POD})),$$

then the results are given in bold numbers in both tables. Also, whenever Algorithm A dominates the standard deviation results of Algorithm B (σ-dominates), i.e.,

$$(\sigma_A(\mathrm{cost}) \leq \sigma_B(\mathrm{cost}) \wedge \sigma_A(\mathrm{POD}) < \sigma_B(\mathrm{POD})) \\ \vee (\sigma_A(\mathrm{cost}) < \sigma_B(\mathrm{cost}) \wedge \sigma_A(\mathrm{POD}) \leq \sigma_B(\mathrm{POD})),$$

then the results are given in bold numbers in both tables.

The cost and POD means of the experiments for the equal business policies are also plotted in Fig. 4 and the means of the experiments for the randomly independent business policies are plotted in Fig. 5.

The empirical study shows that the dynamic program μ-dominates the local strategy for 12 of the 41 business policy experiments whereas the local strategy never μ-dominates the dynamic program. Furthermore, for those equal business policies where the dynamic program does not μ-dominate the local strategy, the local strategy never achieves a POD mean above 50 percent whereas the dynamic program achieves a POD mean of almost 100 percent for most of these business policies with comparable cost means. The same can be observed for the independent business policy experiments. It is also remarkable that even for most of the randomly chosen business policy setups, where the preferences of the decision makers are most diverse, the outcoming POD means of the dynamic program are almost 100 percent. This is a good result, because it means that the algorithms plans robust and coordinates the decision makers.

Now, we observe the expected cost means exclusively. For equal business policies, the mean of the dynamic program is better for 8 setups whereas the mean of the local strategy is better for 10 setups. For randomly chosen

Table 5 Mean and standard deviation for the realized costs and the POD for the two algorithms simulated with 21 equal business policies

q	$\alpha(q)$	Dynamic program		Local strategy		Dynamic program		Local strategy	
		μ(cost)	μ(POD)	μ(cost)	μ(POD)	σ(cost)	σ(POD)	σ(cost)	σ(POD)
0	0	430	0	430	0	0	0	0	0
1	0.05	430	0	430	0	0	0	0	0
2	0.1	439	26	435	0	9	44	**5**	**0**
3	0.15	439	26	439	26	9	44	9	44
4	0.2	439	26	551	49	**9**	**44**	12	50
5	0.25	**467**	**52**	551	49	35	50	**12**	**50**
6	0.3	604	100	558	48	**14**	**7**	20	50
7	0.35	604	100	558	48	**14**	**7**	20	50
8	0.4	604	100	558	48	**14**	**7**	20	50
9	0.45	604	100	558	48	**14**	**7**	20	50
10	0.5	604	100	558	48	**14**	**7**	20	50
11	0.55	604	100	558	48	**14**	**7**	20	50
12	0.6	604	100	558	48	**14**	**7**	20	50
13	0.65	604	100	558	48	**14**	**7**	20	50
14	0.7	604	100	592	48	**14**	**7**	27	50
15	0.75	**604**	**100**	697	67	**14**	**7**	51	47
16	0.8	**604**	**100**	697	67	**14**	**7**	51	47
17	0.85	**604**	**100**	780	93	**14**	**7**	78	26
18	0.9	**604**	**100**	780	93	**14**	**7**	78	26
19	0.95	**720**	**100**	780	93	**0**	**0**	78	26
20	1	**720**	**100**	780	93	**0**	**0**	78	26

business policies, the mean of the dynamic program is better for 5 setups whereas the mean of the local strategy is better for 25 setups. We conclude that the local strategy tends to support the expected costs criterion for randomly chosen business policies.

We exclusively examine the POD means next. For equal business policies, the mean of the dynamic program is better for 17 setups whereas the mean of the local strategy is better for only one setup. For randomly chosen business policies, the mean of the dynamic program is better for 18 setups whereas the mean of the local strategy is never better. Therefore, we conclude that the dynamic program strongly supports the POD probability criterion.

We also conclude from the standard deviation results that the dynamic program is more robust for the planning problem than the local strategy heuristic. The dynamic program σ-dominates the local strategy heuristic 27 times whereas it is only 7 times the other way round. The absolute values of the standard deviation are often significantly smaller for the dynamic program and even 13 times equal to zero.

4.2.2 Application to planning situation

Finally, we present results of how the planning algorithm supports decision making in forwarding air freight. We take the previous evaluation setup, its transport network of Fig. 2, the transport options of Table 4 and the air freight item that should be picked up at the shipper location L_1 at $a_0 = 6$ am and should be delivered at the consignee location L_7 at $t_{\mathrm{POD}} = 5$ pm. The decision maker for the transport from the location L_1 wants to coordinate himself with the possible follow-up decision makers by using the distributed planning algorithm. We assume that all follow-up decision makers follow a business policy of $\alpha_i = 0.5$, $i = 2, \ldots, n$. The first decision maker however does not fix his policy in advance. Instead he wants to compare different nondominated transport options. Therefore, we start the dynamic program for different business policies $\alpha_1 = 0.1q$, $q = 0, 1, \ldots, 10$ of the first decision maker. For each of the eleven resulting business policy setups of the decision maker, we start the dynamic program and gain an evaluation of the transport options of the first decision maker. Table 7 and Fig. 6 illustrate the resulting three nondominated transport options for the first decision maker.

The decision maker who has to decide the first transportation step from the shipper is now able to compare his nondominated transport options that correspond to different business policies he can choose. Each evaluated transport option is aligned with the entire D2D transport network. In this way, the decision maker has a feedback of possible follow-up decisions. For getting these results with the

Table 6 Mean and standard deviation for the realized costs and the POD for the two algorithms simulated with 20 randomly independent business policies

q	Dynamic program		Local strategy		Dynamic program		Local strategy	
	μ(cost)	μ(POD)	μ(cost)	μ(POD)	σ(cost)	σ(POD)	σ(cost)	σ(POD)
0	611	93	601	48	31	25	29	50
1	604	100	488	26	14	7	11	44
2	720	100	558	48	**0**	**0**	20	50
3	604	100	488	26	14	7	11	44
4	720	100	558	48	**0**	**0**	20	50
5	720	100	646	66	**0**	**0**	49	47
6	467	52	439	26	35	50	**9**	**44**
7	**467**	**52**	551	49	35	50	**12**	**50**
8	604	100	601	48	**14**	**7**	29	50
9	720	100	558	48	**0**	**0**	20	50
10	720	100	637	0	**0**	**0**	47	0
11	720	100	558	48	**0**	**0**	20	50
12	**604**	**100**	646	66	**14**	**7**	49	47
13	**467**	**52**	551	49	35	50	**12**	**50**
14	467	52	439	26	35	50	**9**	**44**
15	**430**	**0**	542	0	**0**	**0**	18	0
16	617	99	488	26	20	9	11	44
17	**430**	**0**	542	0	**0**	**0**	18	0
18	720	100	488	26	**0**	**0**	11	44
19	467	52	435	0	35	50	**5**	**0**

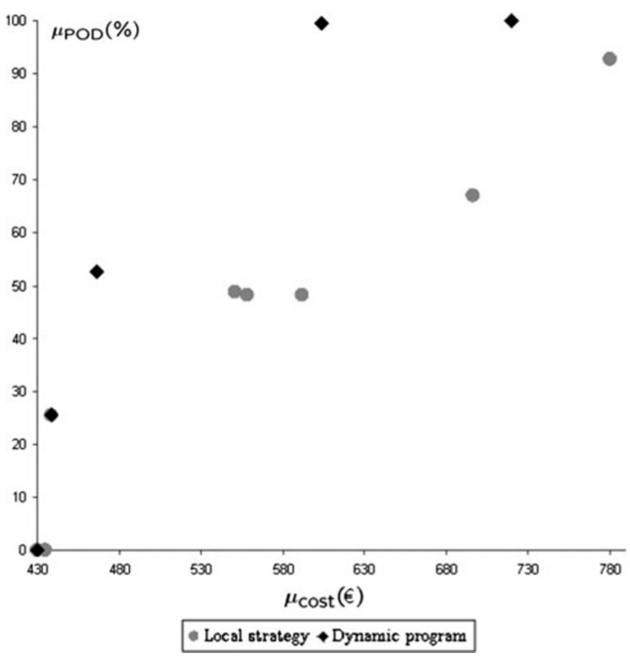

Fig. 4 Cost and POD means of the experiments for the 21 equal business policies

dynamic program he does not have to share all detailed information with a central planning instance and maintains his authority.

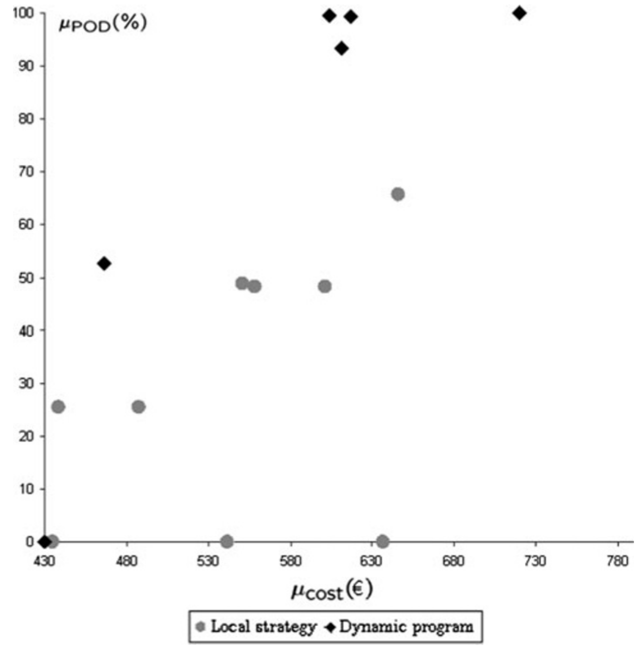

Fig. 5 Cost and POD means of the experiments for the 20 independent business policies

Table 7 Expected costs and POD probability of the three nondominated transport options

T_{ij}^{k}	$C(1,a_0)$	$P(1,a_0)$
T_{12}^{2}	602	98
T_{12}^{3}	465	50
T_{14}^{1}	720	99

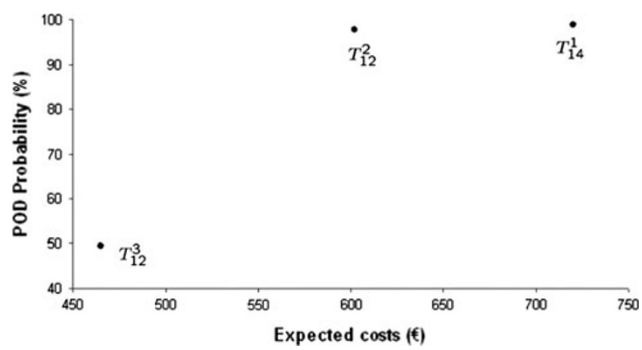

Fig. 6 Nondominated transport options for the first transportation step from the shipper

5 Conclusions

In this paper, we presented initial results of providing decentralized decision support for forwarding air cargo freight along the door to door process chain. We proposed a dynamic programming approach that enables the decision makers to coordinate transportation with the follow-up decision makers in the process chain. Our algorithm needs only lean information exchange that does not reveal sensitive data of the collaborating logistics companies. Each participant can plan his decision locally and the algorithm coordinates the local planning options of the participants. Therefore, it can operate decentralized planning situations. We also compared our approach to a local strategy algorithm and conclude that the dynamic program is more robust for planning in order to deliver on time.

In future, research we refine our decentralized decision support approach: we will extend the approach for other criteria that are important for the D2D process like a Carbon dioxide footprint predictor of the transportation and a utilization measure of the warehouses and hubs; we will replace the binomial distribution of the lateness of the transport options by more detailed probability distributions. We will compare it to other planning algorithms, e.g., an adapted multiobjective stochastic shortest path problem solver or to a stochastic programming approach. We will also test our approach on real world transport networks and transport data in order to substantiate the practical relevance of our work.

Acknowledgments We are indebted to two anonymous referees for their insightful comments on drafts of this paper and for pointing out additional references.

References

1. Abram J, Rhodes I (1982) Some shortest path algorithms with decentralized information and communication requirements. IEEE Trans Autom Control 27(3):570–582. doi:10.1109/tac.1982.1102987
2. Azevedo JA, Costa MEOS, Madeira JJERS, Martins EQV (1993) An algorithm for the ranking of shortest paths. Eur J Oper Res 69(1):97–106. doi:10.1016/0377-2217(93)90095-5
3. Bartodziej P, Derigs U, Zils M (2007) O&D revenue management in cargo airlines—a mathematical programming approach. OR Spectr 29(1):105–121. doi:10.1007/s00291-005-0019-y
4. Bertsekas DP, Tsitsiklis JN (1991) An analysis of stochastic shortest path problems. Math Oper Res 16(3):580–595. doi:10.1287/moor.16.3.580
5. Cho J, Kim H, Choi H (2010) An intermodal transport network planning algorithm using dynamic programming—a case study: from Busan to Rotterdam in intermodal freight routing. Appl Intell 1–13. doi:10.1007/s10489-010-0223-6
6. Cormen TH, Leiserson CE, Rivest RL, Stein C (2001) Introduction to algorithms. 2nd edn. MIT Press, Cambridge
7. Coutinho-Rodrigues JM, Clímaco JCN, Current JR (1999) An interactive bi-objective shortest path approach: searching for unsupported nondominated solutions. Comput Oper Res 26:789–798. doi:10.1016/s0305-0548(98)00094-x
8. Crainic T (2003) Handbook of transportation science. In: International Ser Oper Res Management Sci, vol 56, 2nd edn. Chap. Long Haul Freight Transportation. Springer, New York, pp 451–516
9. Desrochers M, Soumis F (1988) A generalized permanent labelling algorithm for the shortest path problem with time windows. IFOR 26(3):191–212
10. Duffie NA (2008) Challenges in design of heterarchical controls for dynamic logistic systems. In: Kreowski HJ, Scholz-Reiter B, Haasis HD (eds) Dynamics in logistics. Springer, Berlin, pp 3–24. doi:10.1007/978-3-540-76862-3
11. Grünert T, Sebastian HJ (2000) Planning models for long-haul operations of postal and express shipment companies. Eur J Oper Res 122(2):289–309. doi:10.1016/s0377-2217(99)00234-9
12. Hülsmann M, Windt K (eds) (2007) Understanding autonomous cooperation & control in logistics—the impact on management, information and communication and material flow. Springer, Berlin
13. Iori M, Martello S, Pretolani D (2010) An aggregate label setting policy for the multi-objective shortest path problem. Eur J Oper Res 207(3):1489–1496. doi:10.1016/j.ejor.2010.06.035
14. Krajewska M, Kopfer H (2009) Transportation planning in freight forwarding companies: Tabu search algorithm for the integrated operational transportation planning problem. Eur J Oper Res 197(2):741–751
15. Murthy I, Sarkar S (1998) Stochastic shortest path problems with piecewise-linear concave utility functions. Manag Sci 44(11):125–136. doi:10.1287/mnsc.44.11.s125
16. Nielsen LR, Pretolani D, Andersen KA (2009) Bicriterion shortest paths in stochastic time-dependent networks. In: Barichard V, Ehrgott M, Gandibleux X, Tkindt V (eds) Multiobjective programming and goal programming, Lecture Notes in Economics and Mathematical Systems. vol 618, Springer, Berlin, pp 57–67
17. Raith A, Ehrgott M (2009) A comparison of solution strategies for biobjective shortest path problems. Comput Oper Res 36:1299–1331. doi:10.1016/j.cor.2008.02.002
18. Reinhardt LB, Pisinger D (2011) Multi-objective and multi-constrained non-additive shortest path problems. Comput Oper Res 38:605–616. doi:10.1016/j.cor.2010.08.003
19. Schönberger J (2005) Operational freight carrier planning: basic concepts, optimization models and advanced memetic algorithms. 1st edn. Springer, Berlin
20. Tarapata Z (2007) Selected multicriteria shortest path problems: An analysis of complexity, models and adaptation of standard algorithms. Int J Appl Math Comput Sci 17:269–287. doi:10.2478/v10006-007-0023-2
21. Wakuta K (2001) A multi-objective shortest path problem. Math Meth Oper Res 54:445–454. doi:10.1007/s001860100169
22. Wu Y (2010) Modeling of containerized air cargo forwarding problems under uncertainty. J Oper Res Soc (to appear). doi:10.1057/jors.2010.84