



Solving the bicriterion routing and scheduling problem for hazardous materials distribution

Konstantinos N. Androutopoulos, Konstantinos G. Zografos *

Transportation Systems and Logistics Laboratory, Department of Management Science and Technology, Athens University of Economics and Business, Evelpidon 47A and 33, Lefkados 113 62, Athens, Greece

ARTICLE INFO

Article history:

Received 7 January 2009

Received in revised form 14 October 2009

Accepted 5 December 2009

Keywords:

Routing

Dynamic programming

Multiple criteria analysis

Hazardous materials transportation

k-Shortest path

ABSTRACT

Hazardous materials routing and scheduling decisions involve the determination of the minimum cost and/or risk routes for servicing the demand of a given set of customers. This paper addresses the bicriterion routing and scheduling problem arising in hazardous materials distribution planning. Under the assumption that the cost and risk attributes of each arc of the underlying transportation network are time-dependent, the proposed routing and scheduling problem pertains to the determination of the non-dominated time-dependent paths for servicing a given and fixed sequence of customers (intermediate stops) within specified time windows. Due to the heavy computational burden for solving this bicriterion problem, an alternative algorithm is proposed that determines the *k*-shortest time-dependent paths. Moreover an algorithm is provided for solving the bicriterion problem. The proximity of the solutions of the *k*-shortest time-dependent path problem with the non-dominated solutions is assessed on a set of problems developed by the authors.

© 2009 Elsevier Ltd. All rights reserved.

1. Introduction

It is widely recognized that hazardous materials transportation risk may be drastically reduced by planning routes that minimise the accident probability and/or the expected consequences of an accident (List et al., 1991). Significant research has been focused on developing mathematical models and algorithms for determining safe and economical hazardous materials routes. The major part of the relevant research has been focused on routing decisions for transporting shipments of hazardous materials from an origin to a single destination minimising cost and/or risk related criteria. Recently, an alternative direction of research has arisen that is focused on the problem of designing the underlying hazardous materials transportation network (from the perspective of the public authorities) in order to minimise the network-wide risk. This approach is applicable to countries or regions (e.g. Alberta Canada) where the public authorities are entitled to exercise traffic control measures for banning hazardous materials shipments passing through specific parts of the roadway network where various sensitive facilities are located. An extensive literature review for the above type of problems is provided in (Erkut et al., 2007).

Although substantial research effort has been focused on routing decision for the full truckload hazardous materials transportation, limited research has been placed on planning routes for the less-than-truckload hazardous materials transportation. A hazardous materials distribution route involves the service of a set of intermediate stops (customers) usually located in urban or suburban areas. In general, planning truck routes for freight distribution includes the following categories of decisions: (i) the determination of the sequence of stops serviced by each truck and (ii) the specification of the path followed by each truck (route-path) on the underlying roadway network in order to deliver the relevant orders. The latter deci-

* Corresponding author. Tel.: +30 210 82036735; fax: +30 210 8203684.
E-mail address: translog@aub.gr (K.G. Zografos).

sion is closely dependent on the former, since for each given sequence of stops the optimum path traversing the entire sequence of stops is envisaged. However, in the case of hazardous materials deliveries, determining the sequence of stops per truck usually involves various case-specific constraints (Caramia et al., 2007; Avella et al., 2004; Ronen, 1995; Brown et al., 1987; Brown and Graves, 1981). For instance, in the problem of delivering orders for multiple liquefied petroleum products (e.g. various types of gasoline and diesel) which cannot be mixed, the loading constraints constitute a significant criterion for specifying the sequence of visiting the customers (Brown et al., 1987). In this context, the problem of sequencing the stops serviced by a truck depends on the specific substances included in the corresponding orders and the associated (regulatory or operational) loading constraints, thus making it difficult to develop a model for the general case. However, assuming that the sequence of stops for a truck is known in advance, then the routing and scheduling problem reduces to a path-finding problem from an origin to a destination (not necessarily different from the origin) with a set of fixed mandatory intermediate stops. The focus of this paper is on analysing and solving the emerging path-finding problem for a given hazardous materials route. A major feature of the emerging hazardous materials path-finding problem, is that it involves multiple time-dependent routing criteria including the transportation cost expressed by the travel time and the transportation risk. In practice, a significant part of the underlying roadway network covers urban areas, where the travel speeds on the arcs of the network are assumed dynamic (Miller-Hooks and Mahmassani, 2000; Pattanamekar et al., 2003). Furthermore, the dynamic traffic control measures applied by the public authorities for diverting hazardous materials shipments from sensitive facilities (hospitals, schools, shopping centres), imply that the travel time on such a roadway segment depends on the departure time from the upstream node. For instance, many countries are practicing curfews, e.g., time constraints for hazardous materials shipments on certain segments of the roadway network (Cox and Turnquist, 1986) in order to protect any local sensitive facilities (e.g. tunnels) from exposition to the associated risk. This type of practices implies that parts of the network are available only within specific time windows, thus strengthening the assumption on the dynamic nature of travel times.

Recent studies claim that risk attributes like the population exposure and the accident probability may vary considerably for different parts of a day. For instance, it has been observed that the accident probabilities are higher at night. Moreover, the expected population exposure at risk also varies with time since the population density at urban areas changes with the daily activities and the associated mobility of the residents (Erkut and Alp, 2007).

Assuming time-dependent travel time and risk values on each arc of the network, the proposed problem is formulated as a bicriterion shortest time-dependent path problem with fixed mandatory intermediate stops constrained within specified service time windows. It should be clarified that by assuming time-dependent cost (risk and travel time) attributes for the arcs of the underlying network, any solution to the above routing problem cannot be solely identified by a path. Any changes in the schedule of traversing a path might lead to different cost values. Thus, any solution to the above path-finding problem involves the path enhanced with the corresponding departure time from each node included in the path. Each solution of this type will be referred to as a scheduled path.

The proposed bicriterion routing and scheduling problem is NP-Complete since it constitutes a generalisation of the classic bicriterion shortest path problem which has been proved to be NP-Complete (Garey and Johnson, 1979; Warburton, 1987). The computational complexity of the proposed problem constitutes an impediment for solving large instances (on real life roadway networks) of the problem. Alternatively, in the decision making context under study, it might be sufficient to determine and assess a set of alternative scheduled paths with marginal deviation from the optimal solution (in terms of travel time). In this perspective, a decision maker could envisage to investigate alternative near optimal solutions with improved risk performance thus capturing the trade-off between risk and travel time in the neighbourhood of the travel time optimal solution. A solution approach has been developed aiming at identifying the k -shortest scheduled paths. The objective of this approach is not to identify the entire set of non-dominated solutions but only a subset of them lying closer to the minimum travel time solution. The proposed solution algorithm extends the work presented in (Androutsopoulos and Zografos, 2008) which deals with the elementary (i.e., without intermediate stops) k -shortest time-dependent path problem. The new algorithm is designed so as to deal with multiple mandatory intermediate stops and discard any equivalent solutions, i.e., scheduled paths sharing the same sequence of nodes, common departure time from the origin, travel time and risk value but different schedule. The proximity of the k -shortest time-dependent paths with the corresponding non-dominated solutions is assessed for a set of problems developed by the authors. A label setting algorithm based on the Decrease Order Time technique (Chabini, 1998) has been developed for identifying the entire set of non-dominated and non-equivalent solutions to each test problem.

The remainder of this paper consists of six sections. Section 2 presents an overview of the literature on hazardous materials routing and scheduling while Section 3 provides the formulation of the proposed problem. Section 4 provides the solution approach for identifying the k -shortest solutions while Section 5 presents the proposed exact solution approach. Both solution approaches have been applied on a set of problems developed by the authors, and their results are presented in Section 6. Finally, Section 7 provides the concluding remarks arising from the work presented in this paper and indicates directions for future research.

2. Previous related work

In the case of non-negative, time-invariant and deterministic arc cost attributes, the relevant bicriterion path-finding problem with mandatory intermediate stops can be addressed by solving each of the arising intermediate bicriterion short-

est paths problem with any of the relevant existing label setting algorithms (Climaco and Martins, 1982; Hansen, 1980). However, a similar approach cannot be adopted for solving the proposed problem in the case of time-dependent cost attributes. In general the Bellman's principle of optimality for multi-criteria shortest path problems cannot be transferred for the case of time-dependent cost attributes. For instance, if a scheduled path from i to j through q departing at time τ_i is non-dominated, then it is not necessary that the part of the path from i to q departing at τ_i is non-dominated as well. Hamacher et al. (2006) provide an illustrative example to prove the above statement. Thus, the proposed path-finding problem cannot be addressed by simply solving the intermediate bicriterion time-dependent shortest path problems between any pair of subsequent stops. To the knowledge of the authors no research effort has been focused on dealing with this problem.

In the existing literature, two major categories of dynamic path-finding problems for the hazardous materials transportation are encountered: (i) dynamic and deterministic and (ii) dynamic and stochastic. In the dynamic and deterministic path-finding problems, the arc travel time and/or any risk attributes are assumed time-dependent, i.e., the travel time and risk on any arc are expressed as functions of the departure time from the upstream node. Assuming time-dependent arc travel times, Nozick et al. (1997) formulated the integrated routing and scheduling problem for hazardous materials transportation as a multi-objective shortest path problem in a network with time-dependent cost attributes. In their study, they adapt the forward label setting algorithm proposed by Cox (1984) for the time-invariant multi-objective shortest path problem, in order to solve the emerging multi-objective routing and scheduling problem. Their algorithmic approach is heuristic since part of the non-dominated solutions may be omitted. In particular, the Cox's algorithm is based on the adaptation of the Bellman's forward optimality principle for the multi-objective shortest path problem, which, as mentioned above, does not hold for a network with time-dependent arc cost attributes (Hamacher et al., 2006; Nozick et al., 1997). Erkut and Alp (2007) enhanced the above integrated routing and scheduling problem by permitting en-route stops. They formulated the emerging routing problem as a constrained shortest path problem in a network with time-dependent arc travel time and risk values. The objective of their formulation was to determine the minimum risk path under the constraint that the total travel time of the path does not exceed a specified threshold. The solution algorithm they propose for the constrained time-dependent shortest path problem is based on the dynamic programming technique introduced in Cai et al. (1997) which utilises the adaptation of the Bellman's backward optimality principle for the multi-criteria shortest time-dependent problem.

On the other hand, the dynamic and stochastic path-finding problems involve arc travel time and/or risk attributes which are assumed random variables with time-dependent probability distribution. Along this line, Hall (1986) modeled the arc travel time as a discrete random variable with time-dependent probability distribution function and presented a branch and bound algorithm for solving the least expected time path problem. In the same context, Miller-Hooks and Mahmassani (1998a,b) adapted the label correcting algorithm for time-dependent shortest path problems by Ziliaskopoulos and Mahmassani (1993) for determining the least possible time path and a lower bound of the probability of occurrence of the least possible travel time. Miller-Hooks and Mahmassani (2000) developed a label correcting algorithm for determining the least expected time path, based on the assumption that the arc travel time random variables are independent. Miller-Hooks and Mahmassani (1998a,b) propose label correcting procedures for determining pareto-optimal paths in a network with stochastic, time varying link cost attributes, by utilising three alternative stochastic dominance criteria. Chang et al. (2005) assumed normal probability distribution functions for the cost attributes on each arc. Under this assumption Chang et al. developed procedures for propagating means and variances along paths and incorporated them in a best first search (Ahuja et al., 1993) solution algorithm for solving the multi-objective dynamic and stochastic shortest path problem.

Finally, an additional line of research involves independent stochastic time-invariant arc cost attributes. Wijeratne et al. (1993) formulate the hazardous materials routing problem from a single origin to a single destination as a stochastic multi-objective shortest path problem. They introduce two criteria of stochastic dominance for comparing the probability distributions of any pair of alternative paths: (i) comparing the mean value and variance of the paths and (ii) comparing the range of the attribute values for which the cumulative probability function of one path overruns the corresponding function of the other. The authors incorporate the above stochastic dominance criteria in the label setting algorithm for bicriteria shortest path problems proposed by Cox (1984).

A major finding from the relevant studies on multi-criteria path-finding problems implies that assuming dynamic and stochastic arc travel time and risk values leads to a broader set of non-dominated solutions than in the case of dynamic and deterministic values. On the other hand, this last observation implies that a heavy computational burden and immense storage facilities would be required identifying the entire set of non-dominated solutions. The work in this paper is based on the assumption that the arc travel time and risk values are deterministic and time-dependent while a set of prespecified mandatory intermediate stops with arrival time windows are involved between the origin and the destination. In particular, the work in this paper extends the work in Nozick et al. (1997) and Erkut and Alp (2007) by assuming intermediate mandatory stops and arrival time windows. In the present work the solutions are further refined by excluding equivalent paths, i.e. solutions with identical paths, departure time, total travel time and risk, which only differ in the waiting time at the nodes.

3. Definition of the problem

Assume that graph $G(N, A)$ denotes a roadway network, where N is the set of nodes, i.e., roadway junctions, origin, destination, and potential intermediate stops, and A the corresponding set of arcs. Each arc (i, j) of the graph is associated

with two non-negative cost attributes, namely travel time and risk. Both attributes are assumed time-dependent, i.e., their value depends on the departure time (τ) from the upstream node (i), denoted by $c_1^t(i, j)$ and $c_2^r(i, j)$ respectively. In graphs with time-dependent cost attributes, as defined above, the classic definition of a path is not sufficient to express the solutions of any shortest path problem. It is evident that traversing the same path for different departure times from the origin or any intermediate node may lead to different total travel time or/and risk value. Thus, it is essential to extend the classic definition of a path as in Definition 1 below, where the time of leaving each node of a path is also included.

Definition 1. Any path in $G(N, A)$ between two nodes i_0 and i_n enhanced with the departure time ($\tau_{i_q}^d$) from any constituent node i_q of the path, is called scheduled path and it is denoted by $p^\tau[i_0, i_n]$ where τ is the earliest possible departure time (ready time) from the origin i_0 .

Any scheduled path may be written as in (1):

$$p^\tau[i_0, i_n] := \left\{ (i_0, i_1)^{\tau_{i_0}^d}, \dots, (i_{q-1}, i_q)^{\tau_{i_{q-1}}^d}, (i_q, i_{q+1})^{\tau_{i_q}^d}, \dots, (i_{n-1}, i_n)^{\tau_{i_{n-1}}^d} \right\} \quad (1)$$

where $(i_q, i_{q+1})^{\tau_{i_q}^d}$ denotes the arc (i_q, i_{q+1}) of the path $p^\tau[i_0, i_n]$ traversed at time $\tau_{i_q}^d$. The departure time ($\tau_{i_q}^d$) from any constituent node i_q of a scheduled path that includes arc (i_q, i_{q+1}) , takes values on a discrete time horizon $[0, T]$ and satisfies constraint (2).

$$\tau_{i_q}^d + c_1^{\tau_{i_q}^d}(i_q, i_{q+1}) \leq \tau_{i_{q+1}}^d, \quad (i_q, i_{q+1})^{\tau_{i_q}^d} \in p^\tau[i_0, i_n] \quad (2)$$

Inequality (2) implies that the departure from any node (apart from the destination) of a scheduled path should occur following to the arrival at it. The arrival time ($\tau_{i_q}^a$) at any node i_q of a scheduled path is defined by (3).

$$\tau_{i_q}^a := \tau_{i_{q-1}}^d + c_1^{\tau_{i_{q-1}}^d}(i_{q-1}, i_q), \quad (i_{q-1}, i_q)^{\tau_{i_{q-1}}^d} \in p^\tau[i_0, i_n] \quad (3)$$

The path-finding problem presented in this paper involves an origin $s_0 \in N$, a destination s_{n+1} (not necessarily different from s_0) and a sequence of n intermediate stops of a given route R located at the nodes in $R = \{s_1, \dots, s_n\} \subseteq N$. The departure from the origin is constrained within a time window $[a_{s_0}^e, a_{s_0}^l]$ while the arrival at the destination is also constrained within $[a_{s_{n+1}}^e, a_{s_{n+1}}^l]$. Unbounded waiting is assumed on any node of the network, in the sense that each node of the network is associated with a rest area where the truck could temporarily park. However no waiting (or bounded waiting time) might apply for the intermediate stops, i.e. the truck is not allowed to wait at the premises of the relevant customer, although unbounded waiting time might be allowed in the neighbourhood expressed through the relevant node of the network. In order to resolve this issue, the underlying network is slightly modified by creating a dummy node s'_q for any node hosting an intermediate stop s_q . The set A is enhanced with the dummy arcs (s_q, s'_q) and (s'_q, s_q) each one having zero travel time and risk. In this context, the stops are assumed located on the new dummy nodes, while the original nodes are treated as any other node of the network. Thus, unbounded waiting time is allowed on the original nodes while waiting at an intermediate stop s'_q is allowed only until time $a_{s_q}^l$ is reached. However, in the remaining of the paper the notation used for the intermediate stop remains the same, i.e. s_q .

Each intermediate stop (s_q) is associated with a service time (t_{s_q}) and a given service time window denoted by $[a_{s_q}^e, a_{s_q}^l]$, where $a_{s_q}^e$ denotes the earliest time that service at stop s_q may start and $a_{s_q}^l$ denotes the corresponding latest service start time. A truck is not allowed to arrive later than $a_{s_q}^l$, while if it arrives earlier than $a_{s_q}^e$ then the service of the relevant customer has to wait until time $a_{s_q}^e$ is reached. This is a common feature in freight distribution problems where each customer requests from the carrier to arrive at the delivery location within a specified time window. Moreover, the service time window for any intermediate stop s_q imposes implicitly a departure time window $[d_{s_q}^e, d_{s_q}^l]$ as defined in (4) and (5)

$$d_{s_q}^e := a_{s_q}^e + t_{s_q} \quad (4)$$

$$d_{s_q}^l := a_{s_q}^l + t_{s_q} \quad (5)$$

Formulae (4) implies that the departure from the stop s_q could not occur earlier than the earliest possible completion time of the service at that stop. Similarly, formulae (5) implies that the departure from s_q could not occur later than the latest possible completion time of the service at that stop.

The objective of the proposed bicriterion routing and scheduling problem involves the determination of the scheduled path from the origin s_0 to the destination s_{n+1} passing through the mandatory intermediate stops in R respecting the service time windows and minimising the total travel time and the total risk. The above special type of scheduled paths are named scheduled route-paths (since they traverse the entire truck route), and they are denoted as in (6).

$$p^\tau[s_0, R, s_{n+1}] = \{p^{\tau_0}[s_0, s_1], p^{\tau_1}[s_1, s_2], \dots, p^{\tau_{n-1}}[s_{n-1}, s_n], p^{\tau_n}[s_n, s_{n+1}]\} \quad (6)$$

where each $p^\tau[s_q, s_{q+1}]$ denotes an intermediate scheduled path.

The total travel time and risk metrics calculated for each alternative scheduled route-path are given by (7) and (8).

$$c_1(p^{\tau_0}[s_0, R, s_{n+1}]) := \sum_{(i,j) \in p} (c_1^{\tau}(i,j) + t_i^w) \quad (7)$$

$$c_2(p^{\tau_0}[s_0, R, s_{n+1}]) = \sum_{(i,j) \in p} \pi_{ij}^{\tau} C_{ij}^{\tau} \quad (8)$$

where $p \equiv p^{\tau_0}[s_0, R, s_{n+1}]$, t_i^w is the waiting time at node i , π_{ij}^{τ} is the probability of accident on arc (i, j) traversed at time τ , and C_{ij}^{τ} denotes the measure of expected consequences arising in case of an accident occurring at arc (i, j) at time τ .

The metric (7) expresses the total en-route duration of the relevant sequence of the constituent scheduled paths including the waiting at the intermediate stops. Concerning the risk metric, several measures are confronted in the relevant literature, e.g., population exposure (Batta and Chiu, 1988), incident probability (Saccomanno and Chan, 1985), perceived risk (Abkowitz et al., 1990), and expected disutility (Erkut and Ingolfsson, 2000). The present research has adopted the perceived risk measure given by (8).

A major feature of the above bicriterion time-dependent shortest path problem is that it aims to determine the set of non-dominated scheduled route-paths based on the following definition: $p_1^{\tau}(s_0, R, s_{n+1})$ is non-dominated for time τ if and only if there does not exist a scheduled route-path $p_2^{\tau}(s_0, R, s_{n+1})$, $p_2^{\tau}(s_0, R, s_{n+1}) \neq p_1^{\tau}(s_0, R, s_{n+1})$ with $c_{\rho}(p_1^{\tau}(s_0, R, s_{n+1})) \geq c_{\rho}(p_2^{\tau}(s_0, R, s_{n+1})) \forall \rho$ and $c_{\rho_1}(p_1^{\tau}(s_0, R, s_{n+1})) > c_{\rho_1}(p_2^{\tau}(s_0, R, s_{n+1}))$ for at least one $\rho_1 \in \{1, 2\}$. Note that any two scheduled route-paths solving the above problem which depart from the origin at the same time, share the same path (i.e. sequence of nodes), and have equal total travel time and risk, they are non-dominated. However, identifying this type of non-dominated solutions does not provide any added value in the specific decision making process, since the only difference among them pertains to the waiting times at the nodes of their (common) path. The above type of solutions is defined below.

Definition 2. Any two scheduled paths $p_1^{\tau}(i_0, i_n)$ and $p_2^{\tau}(i_0, i_n)$ which share a common path from i_0 to i_n departing from node i_0 at time τ and $c_1(p_1^{\tau}[i_0, i_n]) = c_1(p_2^{\tau}[i_0, i_n])$, $c_2(p_1^{\tau}[i_0, i_n]) = c_2(p_2^{\tau}[i_0, i_n])$, they are called equivalent. Any two scheduled route-paths $p_1^{\tau}[i_0, R, i_n]$ $p_2^{\tau}[i_0, R, i_n]$ are non-equivalent if there is atleast one pair of constituent scheduled paths $p_1^{\tau}[i_{s_q}, i_{s_{q+1}}]$ and $p_2^{\tau}[i_{s_q}, i_{s_{q+1}}]$, which are non-equivalent as well.

In this context, this paper deals with the determination of the non-equivalent solutions of the proposed bicriterion routing and scheduling problem.

4. Determining the k -shortest non-equivalent scheduled route-paths

The proposed solution approach starts from the destination and traverses backwards the stops of the route iteratively, calculating the k -shortest scheduled paths from an intermediate stop s_q to the destination s_{n+1} through the stops $\{s_{q+1}, \dots, s_n\}$ for any possible departure time. Thus, in essence the k -shortest scheduled route-path problem is decomposed to a series of nested k -shortest scheduled path problems. A label setting algorithm has been developed in order solve each of the above problems. The proposed label setting algorithm is based on the work presented in (Androutsopoulos and Zografos, 2008) where the elementary k -shortest time-dependent path problem is addressed under the assumption of unbounded waiting times at the nodes of the network. This algorithm is further enhanced for identifying only the k -shortest non-equivalent solutions. A theoretical discussion is included in the remainder of this section in order to justify this enhancement.

4.1. Previous related work on the k -shortest path problem

In general, the k -shortest path problem arises in many routing decisions involving the determination of more than one alternative paths. The objective of the classic k -shortest path problem is to determine the first k -minimum cost paths in a network where each arc is associated with a static cost attribute. Depending on whether cycles are allowed or not in the paths, two different versions of the problem arise. The research studies by Eppstein (1998) and Martins (1984) were focused on the k -shortest looping path problem. On the other hand, several studies have been focused on the k -shortest loopless paths including Yen (1971), Katoh et al. (1982), and Hadjiconstantinou and Christofides (1999). This paper studies an extension of the classic k -shortest path problem involving the determination of k -shortest (loopless) non-equivalent scheduled paths on a network with time-dependent travel times.

Although substantial research effort has been devoted to the time-dependent shortest path problem (Cooke and Halsey, 1966; Ziliaskopoulos and Mahmassani, 1993; Cai et al., 1997; Chabini, 1998; Ahuja et al., 2003) limited research has been found in the literature addressing the k -shortest time-dependent path problem. Miller-Hooks and Mahmassani (1998a,b) have studied the k -shortest path problem in a network with time-dependent arc travel times. They propose a dynamic programming algorithm founded on the label correcting technique of Ziliaskopoulos and Mahmassani (1993) for determining at most k -shortest paths. Their approach is heuristic in the sense that eventually only $\kappa (< k)$ of the k -shortest looping paths are identified, without specifying how close is κ to k . Other relevant work in this field relates to the k -shortest path problem on networks with time-invariant cost attributes, where either the departure or the arrival at any node is time constrained. In

particular, Chen et al. (2001) studied the k -minimum cost paths problem in a time-schedule network. The objective of their formulation is to determine k -minimum cost paths with total travel time less than a threshold value τ . The corresponding network involves arcs with time-invariant travel times and non-negative cost attributes while the departure from any node is allowed only at specified discrete points in time. The solution algorithm proposed by Chen, Rinks and Tang involves the creation of the time expanded version of the network and the implementation of the Yen's algorithm for the classic k -shortest path problem. Along this line, Chen and Tang (1998) proposed an algorithm for solving a variation of the above problem where the departure time from any node is allowed only within specified time windows. In particular, when an arrival occurs at a node of the relevant network, two options are possible: either leave the node immediately if the arrival lies within a given time window or otherwise wait until the next time window is reached. The solution algorithm for this problem follows the structure of Yen's algorithm where instead of determining simple shortest paths, the entire set of alternative scheduled paths with the same total travel time but different waiting times are identified.

It is evident that the abovementioned k -shortest path problems constitute special cases of the problem studied in this paper which involves time-dependent (instead of time-invariant) cost attributes, multiple intermediate mandatory stops, and time windows on the origin, the destination, and the intermediate stops.

4.2. Solution approach

The proposed algorithmic approach consists of $n + 1$ stages indexed by $v \in Q \equiv \{0, \dots, n\}$. At each stage $v \in Q$, the k -shortest (non-equivalent) scheduled paths from any node of the network to the destination s_{n+1} through the v intermediate stops from s_{n-v+1} to s_n for any possible departure time τ , i.e., $\tau \in [d_{s_{n-v}}^e, d_{s_{n-v}}^l]$ for the node s_{n-v} and $\tau \in [d_{s_{n-v+1}}^e, d_{s_{n-v+1}}^l]$ for any other node. Given the k -shortest scheduled paths from s_{n-v} to s_n , the proposed algorithm proceeds to stage $(v + 1)$ where the k -shortest scheduled paths from s_{n-v-1} to s_n through $\{s_{n-v}, s_{n-v+1}, \dots, s_n\}$ are determined. Each of the scheduled paths determined in stage $(v + 1)$ consists of: (i) a sub-path from the origin to the intermediate stop s_{n-v} and (ii) any of the k -shortest scheduled paths from s_{n-v} to s_n (through $\{s_{n-v+1}, \dots, s_n\}$) determined in stage (v) . The algorithm that solves each of the above mentioned nested k -shortest non-equivalent scheduled path problems is based on the algorithm proposed in (Androutsopoulos and Zografos, 2008) for the elementary k -shortest time-dependent path problem. The proposed algorithm enhances the one presented in (Androutsopoulos and Zografos, 2008) on the basis of identifying only non-equivalent solutions and treating multiple intermediate stops between the origin and the destination. It is proved that under the assumption of unbounded waiting, the exclusion of any equivalent scheduled paths does not pose a risk in omitting any of the k -shortest non-equivalent scheduled paths.

The intuition underlying the proposed algorithm is founded on the Proposition 1 presented below. Definitions 3 and 4 presented below are essential for stating and proving Proposition 1.

Definition 3. Assume a scheduled path $p_1^\tau[i_0, i_n] := \{(i_0, i_n)^{\tau_0}, \dots, (i_{q-1}, i_q)^{\tau_{q-1}}, (i_q, i_{q+1})^{\tau_q}, \dots, (i_{n-1}, i_n)^{\tau_{n-1}}\}$. Any scheduled path $p_2^\tau[i_q, i_n] := \{(i_q, i_{q+1})^{\tau_q}, \dots, (i_{n-1}, i_n)^{\tau_{n-1}}\}$ with $\tau_{q-1} + c_1^{\tau_{q-1}}(i_{q-1}, i_q) \leq \tau' \leq \tau_q$ is called scheduled sub-path of $p_1^\tau(i_0, i_n)$.

Definition 4. Any two scheduled paths $p_{r_1}^\tau[i_0, i_n]$ and $p_{r_2}^\tau[i_0, i_n]$ for which $c_1(p_{r_1}^\tau[i_0, i_n]) = c_1(p_{r_2}^\tau[i_0, i_n])$, they are called time-equivalent solutions.

Proposition 1. If a scheduled path $p_1^\tau[i_0, i_n]$ is one of the k -shortest loopless scheduled paths from i_0 to i_n with ready time τ , then its sub-path $p_2^\tau[i_q, i_n]$ is either one of the k -shortest solutions from i_q to i_n at time τ' or it is time-equivalent with one of them.

Proof. It will be proved by contradiction. Assume that there exist at least k scheduled paths $p_{(r)}^\tau[i_q, i_n]$ for which $c_1(p_{r_2}^\tau[i_q, i_n]) > c_1(p_{r_1}^\tau[i_q, i_n])$ for any $r \in \{1, \dots, k\}$. Joining the first part of $p_1^\tau(i_0, i_n)$, i.e., $\{(i_0, i_n)^{\tau_0}, \dots, (i_{q-1}, i_q)^{\tau_{q-1}}\}$, with each of $p_{(r)}^\tau(i_q, i_n)$ creates k new scheduled paths. If the new scheduled paths created as above contain no loop then they have strictly lower total travel time than $p_1^\tau(i_0, i_n)$, leading to a contradiction with the hypothesis of this proposition. If on the other hand any of the k new scheduled paths contains a cycle, then by replacing the cycle with waiting time at the node revisited, leads to a new scheduled path with no cycle and equal travel time. Thus, in either case the findings contradict with the hypothesis that $p_1^\tau[i_0, i_n]$ is one of the k -shortest scheduled paths. \square

Proposition 1 implies that if a sub-path of $p_1^\tau[i_0, i_n]$ is not within the k -shortest scheduled paths, then it will be time-equivalent with the corresponding k th shortest scheduled path.

A label setting algorithm is proposed for solving the corresponding k -shortest nested scheduled path problem arising at each stage $v \in Q$. Each node of the network is initially associated with a list of k vector-labels $\{(\lambda_{(r),c_1}^{\tau,[v]}(i), \lambda_{(r),c_2}^{\tau,[v]}(i)) : r = 1, \dots, k\}$ for every point in time $\tau \in [d_{s_{n-v}}^e, d_{s_{n-v+1}}^l]$ and stage v of the problem, denoted by $\Lambda_i^{\tau,[v]}$. The first attribute $\lambda_{(r),c_1}^{\tau,[v]}(i)$ of each vector-label denotes the total travel time of the r th shortest scheduled path $p_{(r)}^\tau[i, \{s_{n-v+1}, \dots, s_n\}, s_{n+1}]$ which departs from node i no earlier than time τ , while the second attribute of the vector-label denotes the risk value of the corresponding scheduled path. Each vector-label is associated with a set $F_{(r)}^{\tau,[v]}(i)$ which includes the nodes visited by the corresponding scheduled path. In addition, a pointer $\psi_{(r)}^{\tau,[v]}(i)$ is attached to each vector-label, indicating the next vector-label of the

corresponding r th shortest scheduled path. The role of this pointer is on facilitating the backtracking of each of the k -shortest scheduled paths from any node i to the destination.

```

for  $v := 0$  to  $n+1$  do the following
  If  $v = 0$  then
    Initialize the vector-labels as follows:
 $\lambda_{(r)c_{1,2}}^{\tau,[0]}(i) := M \quad i \in N, i \neq s_{n+1}, r \in \{1, \dots, k\}, \tau \in [d_{s_n}^e, a_{s_{n+1}}^l]$ 
 $\lambda_{(1)c_{1,2}}^{\tau,[0]}(s_{n+1}) := 0, \tau \in [d_{s_n}^e, a_{s_{n+1}}^l]$ 
  Else ( $v > 0$ )
    Initialize the vector-labels as follows (setting  $\tau' = \tau + t_{s_{n-v+1}}$ )
 $\lambda_{(r)c_{1,2}}^{\tau,[v]}(i) := M \quad i \in N, i \neq s_{n-v+1}, r \in \{1, \dots, k\}, \tau \in [d_{s_{n-v}}^e, a_{s_{n-v+1}}^l]$ 
 $\lambda_{(r)c_1}^{\tau,[v]}(s_{n-v+1}) := \lambda_{(r)c_1}^{\tau',[v-1]}(s_{n-v+1}) + t_{s_{n-v+1}} + \max\{0, a_{s_{n-v+1}}^l - \tau\},$ 
 $\tau \in [d_{s_{n-v}}^e, a_{s_{n-v+1}}^l], \tau' \in [d_{s_{n-v+1}}^e, a_{s_{n-v+1}}^l], \Lambda_{s_{n-v+1}}^{\tau',[v-1]} \neq \emptyset$ 
 $\lambda_{(r)c_2}^{\tau,[v]}(s_{n-v+1}) := \lambda_{(r)c_2}^{\tau',[v-1]}(s_{n-v+1}),$ 
 $\tau \in [d_{s_{n-v}}^e, a_{s_{n-v+1}}^l], \tau' \in [d_{s_{n-v+1}}^e, a_{s_{n-v+1}}^l], \Lambda_{s_{n-v+1}}^{\tau',[v-1]} \neq \emptyset$ 
 $\lambda_{(r)c_1}^{\tau,[v]}(s_{n-v+1}) := \lambda_{(r)c_1}^{\tau',[v-1]}(s_{n-v+1}) + t_{s_{n-v+1}} + (a_{s_{n-v+1}}^l - \tau),$ 
 $\tau \in [d_{s_{n-v}}^e, a_{s_{n-v+1}}^l], \tau' \in [d_{s_{n-v+1}}^e, a_{s_{n-v+1}}^l],$ 
 $\Lambda_{s_{n-v+1}}^{\tau',[v-1]} = \emptyset, \tau'' := \min\{\tau : \Lambda_{s_{n-v+1}}^{\tau',[v-1]} \neq \emptyset\}$ 
 $\lambda_{(r)c_2}^{\tau,[v]}(s_{n-v+1}) := \lambda_{(r)c_2}^{\tau',[v-1]}(s_{n-v+1}), \tau \in [d_{s_{n-v}}^e, a_{s_{n-v+1}}^l], \tau' \in [d_{s_{n-v+1}}^e, a_{s_{n-v+1}}^l],$ 
 $\Lambda_{s_{n-v+1}}^{\tau',[v-1]} = \emptyset, \tau'' := \min\{\tau : \Lambda_{s_{n-v+1}}^{\tau',[v-1]} \neq \emptyset\}$ 
  for  $\tau = a_{s_{n-v+1}}^l \rightarrow d_{s_{n-v}}^e$  do the following
  for  $i \in N \setminus \{s_{n-v+1}\}$  do the following
  if ( $i \neq s_{n-v}$ ) or ( $i = s_{n-v}$  and  $\tau \in [d_{s_{n-v}}^e, a_{s_{n-v}}^l]$ ) then do the following
    for  $j \in \Gamma^{+1}(i)$  do the following
      set  $\tau' := \tau + c_1^{\tau}(i, j)$ 
      if  $\tau' \leq a_{s_{n-v+1}}^l$  then do the following
        for  $r := 1 \rightarrow |\Lambda_j^{\tau',[v]}|$  do the following
          If ( $\lambda_{(r)c_{1,2}}^{\tau',[v]}(j) < M$ ) and ( $i \notin F_{(r)}^{\tau',[v]}(j)$ ) then
            Set  $\varphi_{c_1}^{\tau}(i) := \lambda_{(r)c_1}^{\tau',[v]}(j) + c_1^{\tau}(i, j)$ , Set  $\varphi_{c_2}^{\tau}(i) := \lambda_{(r)c_2}^{\tau',[v]}(j) + c_2^{\tau}(i, j)$ 
            If  $\varphi_{(r)c_{1,2}}^{\tau,[v]}(i)$  is non-equivalent with any of the existing vector-labels in  $\Lambda_i^{\tau,[v]}$ 
              Set  $g := 1$  and stop:=false
            While (not stop) and ( $g \leq |\Lambda_i^{\tau,[v]}|$ ) do the following
              If ( $\varphi_{c_1}^{\tau}(i) < \lambda_{c_1}^{\tau,[v]}(i)$ ) then do the following:
                For  $w := |\Lambda_i^{\tau,[v]}| \rightarrow g$  do the following
                  Set ( $\lambda_{(w)c_1}^{\tau,[v]}(i), \lambda_{(w)c_2}^{\tau,[v]}(i)$ ):=( $\lambda_{(w-1)c_1}^{\tau,[v]}(i), \lambda_{(w-1)c_2}^{\tau,[v]}(i)$ )
                  Set ( $\lambda_{(g)c_1}^{\tau,[v]}(i), \lambda_{(g)c_2}^{\tau,[v]}(i)$ ):=( $\varphi_{c_1}^{\tau}(i), \varphi_{c_2}^{\tau}(i)$ ), Set
                   $\psi_{(g)}^{\tau}(i) \rightarrow (\lambda_{(r)c_1}^{\tau,[v]}(j), \lambda_{(r)c_2}^{\tau,[v]}(j))$ 
                  If ( $g=k$ ) or ( $g < k$  and  $\lambda_{(k-1)c_1}^{\tau,[v]} < \lambda_{(k)c_1}^{\tau,[v]}$ ) then exclude from the list
                   $\Lambda_i^{\tau,[v]}$  all vector-labels for  $g > k$ 
                  Set stop:=true
                   $g := g + 1$ 
                If ( $\varphi_{c_1}^{\tau}(i) = \lambda_{c_1}^{\tau,[v]}(i)$ ) and ( $g=k$ ) then
                  Create a new label ( $\lambda_{(\Lambda_i^{\tau,[v]}+1)c_1}^{\tau,[v]}(i), \lambda_{(\Lambda_i^{\tau,[v]}+1)c_2}^{\tau,[v]}(i)$ ):=( $\varphi_{c_1}^{\tau}(i), \varphi_{c_2}^{\tau}(i)$ ),
                  add it to list  $\Lambda_i^{\tau,[v]}$ 
                  Set  $\psi_{(g)}^{\tau}(i) \rightarrow (\lambda_{(r)c_1}^{\tau,[v]}(j), \lambda_{(r)c_2}^{\tau,[v]}(j))$ 
            End of while statement

```

Fig. 1. Steps of the algorithm for solving the k -shortest scheduled path problem.

For the stage $v = 0$, where the k -shortest scheduled path problem from s_n to s_{n+1} for any departure time $\tau \in [d_{s_n}^e, d_{s_n}^l]$ is solved, the proposed algorithm initializes the vector-labels associated with the destination (s_{n+1}) by setting both attributes equal to zero. The attributes of the vector-labels associated with any other node are set equal to a very large number M . For any other stage $v \neq 0$ the travel time and risk attributes of the initial vector-labels associated with the node s_{n-v+1} for time $\tau \in [d_{s_{n-v+1}}^e, d_{s_{n-v+1}}^l]$ are set equal to the value of travel time and the risk value of the corresponding vector-labels for stage $(v - 1)$ at time $\tau' := \tau + t_{s_{n-v+1}}$. A vector-label is also created for the node s_{n-v+1} for any time $d_{s_{n-v}}^e \leq \tau \leq d_{s_{n-v+1}}^e$ by setting the travel time equal to the corresponding travel time of the earliest vector-label from stage $(v - 1)$ plus the service time $t_{s_{n-v+1}}$ plus the waiting time $(d_{s_{n-v+1}}^e - \tau)$, and the risk equal to the corresponding risk value of the earliest vector-label from stage $(v - 1)$.

Following to the vector-labels initialisation, each stage (v) of the algorithm involves $T(= d_{s_{n-v+1}}^l - d_{s_{n-v}}^e + 1)$ iterations moving backwards in time, starting from time $\tau := d_{s_{n-v+1}}^l$ down to $d_{s_{n-v}}^e$. For each iteration τ the k -shortest scheduled paths from any node i to destination node s_{n+1} through the v intermediate stops $\{s_{n-v+1}, \dots, s_n\}$ departing from i at time τ are determined. Each iteration τ involves the identification of a set of candidate vector-labels for every node $i \neq s_{n-v+1}$. Given any arc $(i, j) \in A$ (outgoing from node i) and $K(\geq k)$ vector-labels already identified for the node j at time $\tau' (= \tau + c_1^{\tau, [v]}(i, j))$, then K candidate vector-labels are identified for node i , referring to the corresponding scheduled paths emerging from joining the arc (i, j) with the scheduled paths associated with the existing vector-labels of node j at time τ' , where τ' is the arrival time through arc (i, j) at node j . The above procedure is repeated for every arc outgoing from node (i) . Thus, if $i \notin F_{(r)}^{\tau, [v]}(j)$ (in order to avoid looping paths), for each vector-label $(\lambda_{(r)c_1}^{\tau, [v]}(j), \lambda_{(r)c_2}^{\tau, [v]}(j)) \in \Lambda_j^{\tau, [v]}$ a new candidate label $(\varphi_{c_1}^{\tau, [v]}(i), \varphi_{c_2}^{\tau, [v]}(i))$ is calculated for node i by adding the travel time and risk of (i, j) to $\lambda_{(r)c_1}^{\tau, [v]}(j)$ and $\lambda_{(r)c_2}^{\tau, [v]}(j)$, respectively. It should be clarified that the search for a loop is performed for the part of the corresponding scheduled path between node i and s_{n-v+1} . Thus, node i may be revisited by the part of the scheduled path after the intermediate stop s_{n-v+1} .

An exemption to the above scanning process arises when $i = s_{n-v}$ for which no scanning is performed for any $\tau \notin [d_{s_{n-v}}^e, d_{s_{n-v}}^l]$, since any such solution contravenes the relevant departure time window constraint. This new candidate vector-label is checked if it is one of the k shortest labels for time τ by comparing it with the existing vector-labels (if any). If the scheduled path associated to the new candidate vector-label is time-equivalent with the existing k th scheduled path (i.e. $\varphi_{c_1}^{\tau, [v]} = \lambda_{(k)c_1}^{\tau, [v]}(i)$) then the vector-label is also included in the list of vector-labels $\Lambda_i^{\tau, [v]}$. Thus, any set $\Lambda_i^{\tau, [v]}$ may include more than k vector-labels. However, if the scheduled path associated to a candidate vector-label is equivalent with any of the existing scheduled paths associated to the labels in $\Lambda_i^{\tau, [v]}$, then the corresponding candidate vector-label is excluded from further consideration.

Upon termination of the iterations, the lists of k (or more) vector-labels determined for the stop s_{n-v} for each point in time $\tau \in [d_{s_{n-v}}^e, d_{s_{n-v}}^l]$ correspond to the k -shortest scheduled paths. Moreover, upon termination of the entire process ($n + 1$ stages), the k -first vector-labels in $\Lambda_{s_0}^{\tau, [n+1]}$ refer to the k -shortest scheduled paths. If more than k labels are included in $\Lambda_{s_0}^{\tau, [n+1]}$, then k -shortest vector-labels are selected.

The proposed algorithm is provided in the form of pseudocode in Fig. 1.

Lemma 1. Assume that k loopless scheduled paths from node i_1 to i_n departing at time τ are non-equivalent. If the k scheduled paths emerging from adding $(i_0, i_1)^{\tau}$ in front of each of the above scheduled paths arriving at i_1 earlier than τ are loopless then they are non-equivalent as well.

Proof. Assume any k loopless non-equivalent scheduled paths $p_r^{\tau}(i_1, i_n)$ for $r = 1, \dots, k$, where $p_r^{\tau}(i_1, i_n) = \{(i_1, i_{r_1})^{\tau}, \dots, (i_{r_m}, i_n)^{\tau, r_m}\}$. Since the k scheduled paths are non-equivalent, then from the Definition 2 the following statement holds for any pair of them: (i) if they involve the same sequence of nodes then they differ in at least one of the total travel time or the risk and (ii) if they have the same travel time and risk value then each of them includes different sequence of nodes. Based on the above statements, any pair of the arising scheduled paths either differ in at least one cost attribute (if the corresponding initial scheduled paths also differ in at least one attribute) or they have a different sequence of (if the initial scheduled paths do so as well). Thus, in any case the new k scheduled paths are non-equivalent as well. \square

In the above proof, If any of the emerging k scheduled paths contain a loop, then the scheduled path arising from replacing the loop with a waiting time, might be equivalent with any of the remaining $k - 1$ scheduled paths.

Lemma 2. If two scheduled paths $p_1^{\tau}(i_1, i_n)$ and $p_2^{\tau}(i_1, i_n)$ are equivalent, then the corresponding scheduled paths emerging from adding $(i_0, i_1)^{\tau}$ in front of each of them are equivalent as well.

Proof. It derives directly from Definition 2. \square

Remark 1. In the proof of Lemma 1, if any of the k new scheduled paths $p_r^{\tau}(i_0, i_n)$ includes a loop, then the scheduled path arising from replacing the loop with waiting time, might be equivalent with any of the remaining $k - 1$ scheduled paths. Thus in the context of the algorithm, when new scheduled paths are created by scanning a node, one cannot exclude the case that the new loopless scheduled path may be equivalent with any of the scheduled paths created by adding one unit of waiting time at that node. This is the reason why it is imperative for check of non-equivalent paths throughout the iterations of the proposed algorithm. Moreover, based on Lemma 2, by excluding any solution equivalent with any of the already existing solutions does not include any risk of excluding in advance any of the k -shortest non-equivalent solutions.

Proposition 2 (*Proof of correctness*). The vector-labels solutions determined by the proposed algorithm are associated with the k -shortest non-equivalent scheduled paths.

Proof. Based on the steps of the algorithm any scheduled path including a cycle or being equivalent with any of the existing solutions is excluded from further consideration. Actually, any of the scheduled paths that contains a cycle and its value is included within the k minimum, it is implicitly replaced with a similar scheduled path which instead of having the loop, it involves waiting time at the node revisited equal to the duration of the corresponding loop. Therefore, it is sufficient to prove that at any stage ν of the algorithm, the k -shortest scheduled paths from every node i to the destination $s_{\nu+1}$ through $\{s_{\nu-\nu+1}, \dots, s_{\nu}\}$ for any time $\tau \in [d_{s_{\nu-\nu}}^e, d_{s_{\nu-\nu+1}}^l]$ are determined. It can be proved by induction. For stage $\nu := 0$, it is evident that at time $\tau = d_{s_{\nu+1}}^l$ the vector-labels determined for any node are unique and therefore optimal. Assume that from time $\tau = d_{s_{\nu+1}}^l$ down to time $\tau_0 + 1$ the vector-labels identified in $\Lambda_{i_q}^{\tau, [0]}$, for each node i_q and any time τ such that $\tau_0 + 1 \leq \tau \leq d_{s_{\nu+1}}^l$, are the k -shortest. Then it is sufficient to prove that the k vector-labels determined by the algorithm for any node i_q at time τ_0 are indeed the k -shortest for departure time τ_0 . This last statement will be proved by contradiction. Assume that there exists a loopless scheduled path $p^{\tau_0}[i_q, s_{\nu+1}]$ which consists of the following sequence of arcs $\{(i_q, i_{q+1})^{\tau_0}, (i_{q+1}, i_{q+2})^{\tau_0}, \dots, (i_{q+m}, s_{\nu+1})^{\tau_0}\}$ and it is one of the k -shortest non-equivalent paths not identified by the proposed algorithm, i.e., there does not exist a label $(\lambda_{(r)c_1}^{\tau_0, [0]}(i_q), \lambda_{(r)c_2}^{\tau_0, [0]}(i_q)) r = 1, \dots, |\Lambda_{i_q}^{\tau_0, [0]}|$ associated with it, while in addition it is non-equivalent with any of k -shortest scheduled paths with determined by the algorithm. A necessary condition for $p^{\tau_0}[i_q, s_{\nu}]$ being within the k -shortest paths while not associated with any label in $\Lambda_{i_q}^{\tau_0, [0]}$ is that $\exists r : c_1(p^{\tau_0}[i_q, s_{\nu+1}]) \leq \lambda_{(r)c_1}^{\tau_0, [0]}(i_q), r \in \{1, \dots, |\Lambda_{i_q}^{\tau_0, [0]}|\}$. Based on Proposition 1, the sub-path $p^{\tau_0}[i_{q+1}, s_{\nu+1}] := \{(i_{q+1}, i_{q+2})^{\tau_0}, \dots, (i_{q+m}, s_{\nu+1})^{\tau_0}\}$ of $p^{\tau_0}[i_q, s_{\nu+1}]$ is either among the k -shortest for node i_{q+1} at time τ_0 or it is time-equivalent with one of them for node $q + 1$, i.e., $p_{(\theta_1)}^{\tau_0}[i_{q+1}, s_{\nu+1}]$ already determined by the algorithm (based on our hypothesis that the algorithm determines the k -shortest vector-labels for any node and $\tau \geq \tau_0 + 1$). Thus based on the steps of the algorithm, a vector-label for path $p^{\tau_0}(i_q, s_{\nu+1})$ should have been determined for node i_{q+1} for time τ , leading to a contradiction with the hypothesis that the scheduled path $p^{\tau_0}(i_q, s_{\nu+1})$ is not represented in $\Lambda_{i_q}^{\tau_0, [0]}$. Assume that the algorithm identifies the k -shortest scheduled paths for $\nu = \nu_0$. Then by following the same procedure as above it can be proved that the algorithm determines the k -shortest scheduled paths for the stage $\nu = \nu_0 + 1$, thus proving the proposition. \square

5. Determining the set of non-dominated non-equivalent solutions

A label setting algorithm has been developed for identifying the entire set of non-dominated non-equivalent solutions, based on the adaptation of the Bellman's backward optimality principle (Hamacher et al., 2006) for networks with multiple time-dependent non-negative cost attributes. The adapted Bellman's optimality principle implies that if a scheduled path from any node i departing at τ_i to node j passing through node q at time τ_q is non-dominated for time τ_i then the sub-path from q departing at time τ_q is non-dominated for time τ_q .

The proposed algorithmic approach is based on decomposing the bicriterion scheduled path-finding problem with mandatory intermediate stops to a series of bicriterion nested shortest scheduled path problems starting from the destination and traversing the route backwards. Solving each of the $(n + 1)$ bicriterion nested shortest scheduled path problems constitutes a separate stage of the algorithmic approach. In particular, at each stage ν the non-dominated solutions from $s_{\nu-\nu}$ to $s_{\nu+1}$ through stops $\{s_{\nu-\nu+1}, \dots, s_{\nu}\}$ are calculated by enhancing (backwards) the non-dominated non-equivalent scheduled paths from $s_{\nu-\nu+1}$ to $s_{\nu+1}$ through the stops $\{s_{\nu-\nu+2}, \dots, s_{\nu}\}$ for each possible departure time $\tau \in [d_{s_{\nu-\nu}}^e, d_{s_{\nu-\nu}}^l]$. The algorithmic procedure that solves each of the bicriterion nested shortest scheduled path problems is based on the adaptation of the Decreased Order Time (DOT) technique (Chabini, 1998) for the bicriterion case.

For the purposes of the proposed label setting algorithm, each node i of the network is associated with the vector-labels $(\xi_{w,c_1}^{\tau, [v]}(i), \xi_{w,c_2}^{\tau, [v]}(i))$, the pointer $\psi_w^{\tau, [v]}(i)$, and the set $F_w^{\tau, [v]}(i)$, where $\xi_{w,c_1}^{\tau, [v]}(i)$ and $\xi_{w,c_2}^{\tau, [v]}(i)$ denote the total travel time and risk (respectively) of the w th non-dominated scheduled path from node i to node $s_{\nu+1}$ through the stops $\{s_{\nu-\nu+1}, \dots, s_{\nu}\}$ at the ν th stage. The pointer $\psi_w^{\tau, [v]}(i)$ points to the next vector-label in the corresponding scheduled path, while the set $F_w^{\tau, [v]}(i)$ denotes the list of nodes already visited by the above mentioned w th scheduled path. Keeping track of the visited nodes of a scheduled path facilitates the check for tracing and excluding equivalent solutions. In addition, a set $\Xi^{\tau, [v]}(i)$ is created for each node i which includes the corresponding non-dominated vector-labels for node i at time τ and stage ν . Fig. 2 presents the steps of the proposed algorithm for any stage $\nu \in Q := \{0, \dots, n\}$. Note that upon the completion of stage n , the entire set of non-dominated non-equivalent solutions are identified for each possible departure time from the origin s_0 .

The operator \oplus denotes the inclusion of a set of candidate vector-labels in a set of non-dominated non-equivalent vector-labels. This operator implies that each candidate vector-label is assessed in terms of dominating or being dominated by each element of the set of currently non-dominated vector-labels. If the candidate vector-label is dominated by at least one element of this set then it is disregarded. In case that it dominates any of the elements of the set of non-dominated vector-labels, then the corresponding element is excluded from the set. If by the end of this process the candidate vector-label is not found either equivalent with or dominated by any of the elements of the set of the temporarily non-dominated vector-labels, it also becomes an element of this set.

For $v := 0$ to $n + 1$ do the following
 If $v \neq 0$ then
 Given $(\xi_{w,c_1}^{\tau, [v-1]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v-1]}(s_{n-v+1})) \forall w$ and $\tau \in [d_{s_{n-v+1}}^e, d_{s_{n-v+1}}^l]$.
 For $\tau \in [d_{s_{n-v}}^e, d_{s_{n-v+1}}^l]$ do the following:
 If $(\tau + t_{s_{n-v+1}} \leq d_{s_{n-v+1}}^l)$ and $(\exists \xi_{w,c_1}^{\tau+t, s_{n-v+1}, [v-1]})$ then $\forall w$ set:

$$\xi_{w,c_1}^{\tau, [v]}(s_{n-v+1}) := \xi_{w,c_1}^{\tau+t, s_{n-v+1}, [v-1]}(s_{n-v+1}) + t_{s_{n-v+1}}$$

$$\xi_{w,c_2}^{\tau, [v]}(s_{n-v+1}) := \xi_{w,c_2}^{\tau+t, s_{n-v+1}, [v-1]}(s_{n-v+1})$$

$$\psi_w^{\tau, [v]}(s_{n-v+1}) \rightarrow (\xi_{w,c_1}^{\tau, [v-1]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v-1]}(s_{n-v+1}))$$

 If $(d_{s_{n-v+1}}^e > \tau + t_{s_{n-v+1}})$ then $\forall w$ set:

$$\xi_{w,c_1}^{\tau, [v]}(s_{n-v+1}) := \xi_{w,c_1}^{\tau, [v-1]}(s_{n-v+1}) + t_{s_{n-v+1}} + (d_{s_{n-v+1}}^e - \tau - t_{s_{n-v+1}})$$

$$\xi_{w,c_2}^{\tau, [v]}(s_{n-v+1}) := \xi_{w,c_2}^{\tau, [v-1]}(s_{n-v+1}), \quad \tau'' = d_{s_{n-v+1}}^e$$

$$\psi_w^{\tau, [v]}(s_{n-v+1}) \rightarrow (\xi_{w,c_1}^{\tau, [v-1]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v-1]}(s_{n-v+1}))$$

 Else (if $v = 0$)

$$\xi_{w,c_1}^{\tau, [v]}(s_{n-v+1}) := 0$$

$$\xi_{w,c_2}^{\tau, [v]}(s_{n-v+1}) := 0$$

$$\Xi^{\tau, [v]}(s_{n-v+1}) := \Xi^{\tau, [v]}(s_{n-v+1}) \oplus (\xi_{w,c_1}^{\tau, [v]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v]}(s_{n-v+1}))$$
 for $\tau := d_{s_{n-v+1}}^l$ down to $d_{s_{n-v}}^e$ do the following
 for $i \in N \setminus \{s_{n-v+1}\}$ do the following
 if $(i \neq s_{n-v})$ or $(i = s_{n-v}$ and $\tau \in [d_{s_{n-v}}^e, d_{s_{n-v}}^l])$ then do the following
 for $j \in \Gamma^{+1}(i)$ do the following
 Set $\tau' := \tau + c_1^{\tau}(i, j)$
 If $\tau' \leq d_{s_{n-v+1}}^l$ then
 for $w := 1$ to $|\Xi^{\tau', v}(j)|$
 Set $\xi_{w,c_1}^{\tau, [v]}(i) := \xi_{w,c_1}^{\tau', [v]}(j) + c_1^{\tau}(i, j), \quad \xi_{w,c_2}^{\tau, [v]}(i) := \xi_{w,c_2}^{\tau', [v]}(j) + c_1^{\tau}(i, j)$

$$\psi_w^{\tau, [v]}(i) \rightarrow (\xi_{w,c_1}^{\tau, [v]}(j), \xi_{w,c_2}^{\tau, [v]}(j))$$

$$\Xi^{\tau, [v]}(i) := \Xi^{\tau, [v]}(i) \oplus (\xi_{w,c_1}^{\tau, [v]}(i), \xi_{w,c_2}^{\tau, [v]}(i))$$

Fig. 2. Steps of the algorithm for solving the bicriterion scheduled path problem.

On termination of the proposed algorithm, each set $\Xi^{\tau, [v+1]}(s_0)$ includes the vector-labels referring to the entire set of non-dominated non-equivalent scheduled route-paths from s_0 to s_{n+1} through $\{s_1, \dots, s_n\}$ for each possible departure time τ .

Lemma 3. If $\Xi^{\tau, [v]}(s_{n-v+1})$ includes the vectors-labels associated to a set of non-dominated non-equivalent solutions for time τ , then the corresponding vector-labels in $\Xi^{\tau-t_{s_{n-v+1}}, [v+1]}(s_{n-v+1})$ emerging through the proposed algorithm, relate to non-dominated non-equivalent scheduled paths as well.

Proof. The vector-labels in $\Xi^{\tau-t_{s_{n-v+1}}, [v+1]}(s_{n-v+1})$ correspond to non-equivalent scheduled paths since the only difference with the vector-labels in $\Xi^{\tau, [v]}(s_{n-v+1})$ relates to the departure time. Thus, it is sufficient to prove that the vector-labels are non-dominated. For any vector-label $(\xi_{w,c_1}^{\tau, [v+1]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v+1]}(s_{n-v+1})) \in \Xi^{\tau-t_{s_{n-v+1}}, [v+1]}(s_{n-v+1})$ there exist a vector-label $(\xi_{w,c_1}^{\tau, [v]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v]}(s_{n-v+1})) \in \Xi^{\tau, [v]}(s_{n-v+1})$ such that $(\xi_{w,c_1}^{\tau, [v+1]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v+1]}(s_{n-v+1})) := (\xi_{w,c_1}^{\tau, [v]}(s_{n-v+1}) + t_{s_{n-v+1}}^w, \xi_{w,c_2}^{\tau, [v+1]}(s_{n-v+1}))$ where $t_{s_{n-v+1}}^w$ denotes the waiting time at stop s_{n-v+1} . Thus it derives from the above statement that the vector-labels $(\xi_{w,c_1}^{\tau, [v+1]}(s_{n-v+1}), \xi_{w,c_2}^{\tau, [v+1]}(s_{n-v+1}))$ are non-dominated as well. \square

Proposition 3. The proposed algorithm identifies the entire set of non-dominated non-equivalent scheduled paths with intermediates stops for any possible departure time.

Proof. It is sufficient to prove that the proposed algorithm determines the non-dominated non-equivalent solutions for any stage v . This can be proved by induction. For $v = 0$, the relevant problem simplifies to a bicriterion shortest scheduled path

problem from s_n to s_{n+1} . Based on the relevant vector-label initialisation process of the algorithm, the destination s_{n+1} is associated with a single vector-label for every $\tau \in [d_{s_n}^e, a_{s_{n+1}}^l]$ equal to $(0, 0)$, which is trivially non-dominated. To complete the proof, it is sufficient to show that given the vector-labels in $\Xi^{\tau, [v]}(s_{n-v+1})$, the initial vector-labels for node s_{n-v+1} in stage $v + 1$ are non-dominated and non-equivalent. In particular, assume that the sets $\Xi^{\tau, [v]}(s_{n-v+1})$, $\tau \in [d_{s_{n-v+1}}^e, a_{s_{n-v+1}}^l]$ consist of the non-dominated non-equivalent vector-labels for any possible departure time τ from stop s_{n-v+1} to the destination s_{n+1} through the stops $\{s_{n-v+2}, \dots, s_n\}$. Based on Lemma 3, the vector-labels in $\Xi^{\tau, [v+1]}(s_{n-v+1})$ calculated by vector-labels initialisation process of the proposed algorithm are non-dominated and non-equivalent. Assume that $\Xi^{\tau, [v+1]}(j) \forall j$ and any $\tau \geq \tau_0 - 1$ for $\tau_0 \in \{1, \dots, T\}$, contains the entire set of non-dominated non-equivalent vector-labels for time τ and node j . It is profound (from the algorithm outline) that the new vector-labels in the sets $\Xi^{\tau_0, [v+1]}(i)$ are non-dominated and non-equivalent. It suffices to prove that there are no other non-dominated non-equivalent scheduled paths from any node i to node s_{n+1} through $\{s_{n-v+1}, \dots, s_n\}$ departing at time τ_0 . If there was such a scheduled path it would be of the following form: $p^{\tau_0}[i, \{s_{n-v+1}, \dots, s_n\}, s_{n+1}] := \{(i, j)^{\tau_0}, p^{\tau'}[j, \{s_{n-v+1}, \dots, s_n\}, s_{n+1}]\}$, where $p^{\tau'}[j, \{s_{n-v+1}, \dots, s_n\}, s_{n+1}]$ is non-dominated and non-equivalent (by assumption since $\tau' \geq \tau_0 - 1$). However, since $p^{\tau'}[j, \{s_{n-v+1}, \dots, s_n\}, s_{n+1}]$ is determined by the proposed algorithm, the scheduled path $p^{\tau_0}[i, \{s_{n-v+1}, \dots, s_n\}, s_{n+1}]$ should have identified as well through the scanning process of the algorithm. \square

6. Computational performance

The proposed algorithm for determining the k -shortest non-equivalent scheduled route-path problem has been applied on a set of problems developed by the authors. The major goals of assessing the proposed algorithm were to compare the solutions produced by the algorithm for the k -shortest non-equivalent scheduled route-paths with the corresponding non-dominated solutions identified by the algorithm that solves the bicriterion problem.

Three grid-like graphs were developed in this context, i.e., G100 with 100 nodes, G300 with 300 nodes, and G600 with 600 nodes. The in-degree and out-degree of each node of the above graphs is equal to 4, in order to strengthen the networks resemblance to roadway networks. The travel time and risk defined on each arc of the graph were defined on 24 time periods (i.e., with time period length equal to 15 min) starting from 08:00 up to 14:00. The problems generated on graphs G100–300 involved randomly selected origins, destinations, and 1 up to 4 intermediate stops with 1–2 h time windows. In this setting, 100 problems were generated for each combination of graph type and number of intermediate stops. The above problems were solved by both algorithms implemented on a Pentium IV, with Intel processor at 2.99 GHz, and 2 GB RAM.

Table 1 presents the average computational time needed for solving the above set of test problems by the algorithm for the k -shortest scheduled route-path problem. Each of these test problems was solved for k equal to 10 and 20. A major finding of the above process is that reasonable computational time was needed for the solution of the test problems ranging from 2 to 72 s.

Moreover, Figs. 3 and 4 indicate that the relationship between the average computational time of the proposed algorithm with the number of intermediate stops is linear.

Table 1
Average computational time of the algorithm for the test problems.

Network id	k	Intermediate stops	Computational time (ms)
G100	10	1	2355
	20	1	4591
	10	2	4197
	20	2	4185
	10	3	4404
	20	3	8996
	10	4	5749
	20	4	11 903
G300	10	1	6665
	20	1	13 708
	10	2	10 722
	20	2	22 625
	10	3	14 315
	20	3	30 332
	10	4	19 489
	20	4	37 284
G600	10	1	13 303
	20	1	27 029
	10	2	13 302
	20	2	27 029
	10	3	30 251
	20	3	59 175
	10	4	38 306
	20	4	71 361

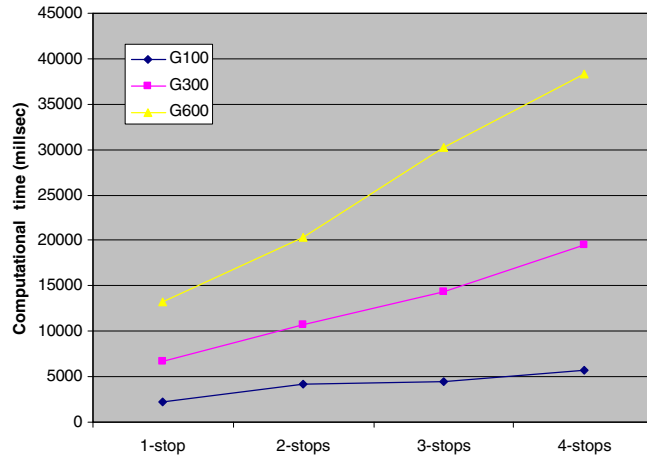


Fig. 3. Average computational time (for $k = 10$) as a function of the number of intermediate stops.

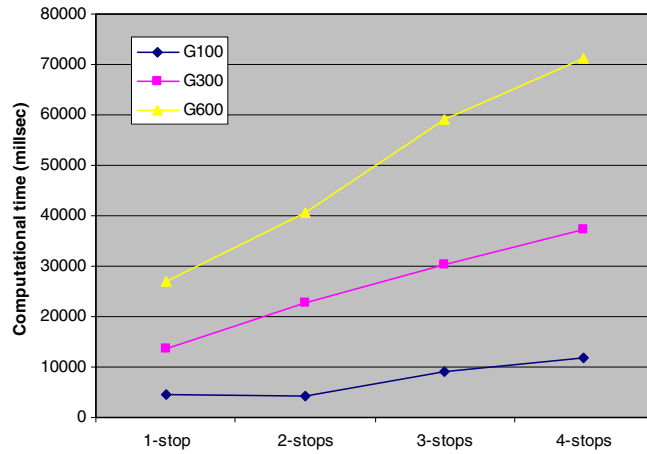


Fig. 4. Average computational time (for $k = 20$) as a function of the number of intermediate stops.

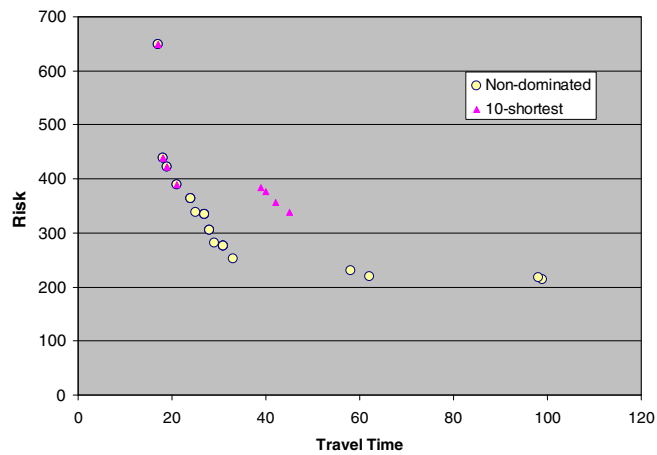


Fig. 5. The non-dominated solutions and the solutions determined by the proposed algorithm (for $k = 10$) plotted on the travel time vs. risk plane.

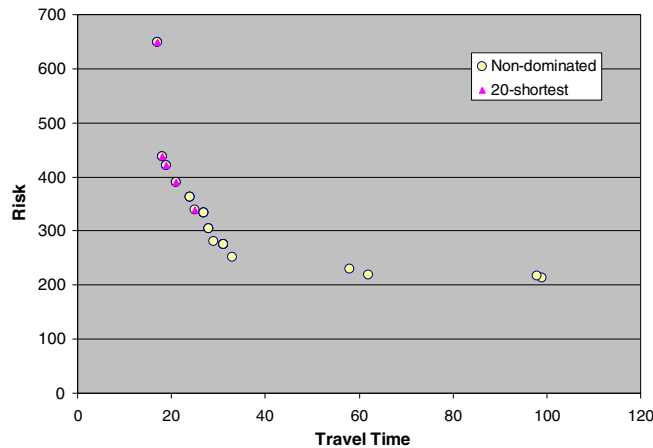


Fig. 6. The non-dominated solutions and the solutions determined by the proposed algorithm (for $k = 20$) plotted on the travel time vs. risk plane.

Concerning the quality of the solutions provided by the algorithm determining the k -shortest scheduled route-paths, Figs. 5 and 6 below presented graphically the deviation of the k -shortest scheduled route-paths from the corresponding non-dominated solutions for one of the test problems mentioned above. Each of the circles plotted in the diagram (travel time vs. risk) of Fig. 5 denotes a non-dominated (non-equivalent) solution to a test problem defined on network G100 with two intermediate stops. The triangles marked on the same plot diagram denote the solutions that emerged from the algorithm that determines the k -shortest scheduled route-paths, for k equal to 10. Note that the solutions plotted in the diagram are only those that were found non-dominated among the total ($k \cdot T$) solutions identified by the algorithm. Similarly, Fig. 6 presents the non-dominated solutions (circles) and the corresponding k -shortest scheduled route-path solutions that emerged for k equal to twenty.

Figs. 5 and 6 indicate that the solutions determined for the k -shortest scheduled route-path problem are close or coincide with the non-dominated solutions lying close to the area of the minimum travel time solution. Moreover, when k increases a better fit of the two types of solutions (non-dominated vs. k -shortest) emerges.

7. Concluding remarks

The bicriterion path-finding problem presented in this paper arises in hazardous materials delivery problems, especially in cases where the sequence of stops per truck is dominated by safety and/or business constraints. In this respect, the sequence of stops is usually specified exogenously, and thus the trade-off between travel time and risk is specified on the basis of the path-finding problem. Moreover, the work presented in this paper could be utilised within any bicriterion vehicle routing and scheduling problem provided that the sequence of stops per route are specified by a separate heuristic algorithm.

An algorithm is presented for determining the non-dominated scheduled route-paths. Alternatively a label setting algorithm is proposed for determining the k -shortest non-equivalent scheduled route-paths, thus approximating the non-dominated solutions close to the minimum travel time solution. The computational performance of the proposed algorithm indicates that reasonable computational time is required for solving real life hazardous materials delivery problems. Therefore, the algorithm could be integrated into applications for planning hazardous materials truck routes given the sequence of the relevant stops. The k -shortest scheduled route-paths provided by the proposed algorithm enable the decision maker to perform a comparative assessment of the alternative solutions with respect to their travel time and risk.

It should be emphasized however, that the proposed algorithm is accurate only under the assumption that unbounded waiting is allowed at any node of the network. Thus, a significant line for future research arises in exploring the above problem under various constraints on the waiting time at the nodes of the network. Moreover, solving the relevant bicriterion vehicle routing and scheduling problem on time-varying networks constitutes another significant direction for future research.

Acknowledgements

The research reported in this paper has been partially supported by the General Secretariat for Research and Technology of the Hellenic Ministry of Development through the Projects SDEF, ENOSIS, and WISETRIP.

References

- Abkowitz, M., Cheng, P.D., Lepofsky, M., 1990. Use of geographic information systems in managing hazardous materials shipments. *Transportation Research Record* 1497, 35–43.
- Ahuja, R., Magnanti, T., Orlin, B., 1993. *Network flows: Theory, Algorithms, and Applications*. Prentice Hall, New Jersey.
- Ahuja, R.K., Orlin, J.B., Pallottino, S., Scutellà, M.G., 2003. Dynamic shortest paths minimizing travel times and costs. *Networks* 41, 197–205.
- Androutsopoulos, K.N., Zografos, K.G., 2008. Solving the k -shortest path problem with time windows in a time varying network. *Operations Research Letters* 36 (6), 692–695.
- Avella, P., Boccia, M., Sforza, A., 2004. Solving a fuel delivery problem by heuristic and exact approaches. *European Journal of Operational Research* 152, 170–179.
- Batta, R., Chiu, S.S., 1988. Optimal obnoxious paths on a network: transportation of hazardous materials. *Operations Research* 36, 84–92.
- Brown, G., Ellis, C.J., Graves, G.W., Ronen, D., 1987. Real-time, wide area dispatch of mobil tank trucks. *Interfaces* 17, 107–120.
- Brown, G., Graves, G.W., 1981. Real-time dispatch of petroleum tank trucks. *Management Science* 27 (1), 19–32.
- Cai, X., Kloks, T., Wong, C.K., 1997. Time-varying shortest path problems with constraints. *Networks* 29 (3), 141–149.
- Caramia, M., Dell'Olmo, P., Gentili, M., Mirchandani, P.B., 2007. Delivery itineraries and distribution capacity of a freight network with time slots. *Computers and Operations Research* 34 (6), 1585–1600.
- Chabini, I., 1998. Discrete dynamic shortest path problems in transportation applications. *Transportation Research Record: Journal of the Transportation Research Board* 1645, 170–175.
- Chang, T.-S., Nozick, L.K., Turnquist, M.A., 2005. Multiobjective path finding in stochastic dynamic networks with application to routing hazardous materials shipments. *Transportation Science* 39 (3), 383–399.
- Chen, Y.L., Rinks, D., Tang, K., 2001. The first K minimum cost paths in a time-schedule network. *Journal of the Operational Research Society* 52, 102–108.
- Chen, Y.L., Tang, K., 1998. Minimum time paths in a network with mixed time constraints. *Computers and Operations Research* 25 (10), 793–805.
- Climaco, J.C.N., Martins, E.Q.V., 1982. A bicriterion shortest path algorithm. *European Journal of Operational Research* 11, 399–404.
- Cooke, K.L., Halsey, E., 1966. The shortest route through a network with time-dependent intermodal transit times. *Journal of Mathematical Analysis and Applications* 14, 493–498.
- Cox, G., 1984. *Routing and Scheduling of Hazardous Materials Shipments: Algorithmic Approaches to Managing Spent Nuclear Fuel Transport*. Ph.D. Thesis, Cornell University, Ithaca, NY.
- Cox, R.G., Turnquist, M.A., 1986. Scheduling truck shipments of hazardous materials in the presence of curfews. *Transportation Research Record* 1063, 21–26.
- Eppstein, D., 1998. Finding the k shortest paths. *SIAM J Computing* 28 (2), 652–673.
- Erkut, E., Ingolfsson, A., 2000. Catastrophe avoidance models for hazardous materials route planning. *Transportation Science* 34 (2), 165–179.
- Erkut, E., Alp, O., 2007. Integrated routing and scheduling of hazmat trucks with stops en route. *Transportation Science* 41 (1), 107–122.
- Erkut, E., Tjandra, S., Verter, V., 2007. Hazardous materials transportation. In: Barnhart, Cynthia, Laporte, Gilbert (Eds.), *Handbooks of Operational Research and Management Science*, vol. 14. pp. 539–621.
- Garey, M., Johnson, D., 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman, San Francisco.
- Hadjiconstantinou, E., Christofides, N., 1999. An efficient implementation of an algorithm for finding K shortest simple paths. *Networks* 34 (2), 88–101.
- Hall, R.W., 1986. The fastest path through a network with random time-dependent travel times. *Transportation Science* 20 (3), 182–188.
- Hansen, P., 1980. Bicriterion path problems. In: Fandel, G., Gal, T. (Eds.), *Multiple Criteria Decision Making: Theory and Applications*, Lecture Notes in Economics and Mathematical Systems, vol. 177. Springer, Heidelberg, pp. 109–127.
- Hamacher, H.W., Ruzika, S., Tjandra, S.A., 2006. Algorithms for time-dependent bicriteria shortest path problems. *Discrete Optimization* 3, 238–254.
- Katoh, N., Ibaraki, T., Mine, H., 1982. Efficient algorithm for k -shortest simple paths. *Networks* 12 (4), 411–427.
- List, G., Mirchandani, P.B., Turnquist, M.A., Zografos, K.G., 1991. Modelling and analysis for hazardous materials transportation: risk analysis, routing/scheduling and facility location. *Transportation Science* 25 (2), 100–114.
- Martins, E.Q.V., 1984. An algorithm for ranking paths that may contain cycles. *European Journal of Operational Research* 18 (1), 123–130.
- Miller-Hooks, E.D., Mahmassani, H.S., 1998a. Least possible time paths in stochastic, time-varying networks. *Computers and Operations Research* 25 (12), 1107–1125.
- Miller-Hooks, E.D., Mahmassani, H.S., 1998b. Optimal routing of hazardous materials in stochastic, time-varying transportation networks. *Transportation Research Record: Journal of the Transportation Research Board* 1645, 143–151.
- Miller-Hooks, E.D., Mahmassani, H.S., 2000. Least expected time paths in stochastic, time-varying transportation networks. *Transportation Science* 34 (2), 198–215.
- Nozick, L.K., List, G.F., Turnquist, M.A., 1997. Integrated routing and scheduling in hazardous materials transportation. *Transportation Science* 31 (3), 200–215.
- Pattanamekar, P., Park, D., Rilett, L.R., Lee, J., Lee, C., 2003. Dynamic and stochastic shortest path in transportation networks with two components of travel time uncertainty. *Transportation Research Part C* 11 (5), 331–354.
- Ronen, D., 1995. Dispatching petroleum products. *Operations Research* 43 (3), 379–387.
- Saccomanno, F., Chan, A.Y., 1985. Economic evaluation of routing strategies for hazardous road shipments. *Transportation Research Record* 12–18.
- Wijeratne, A.B., Turnquist, M.A., Mirchandani, P.B., 1993. Multi-objective routing of hazardous materials in stochastic networks. *European Journal of Operational Research* 65, 33–43.
- Warburton, A., 1987. Approximation of pareto optima in multiple objective, shortest-path problems. *Operations Research* 35 (1), 70–79.
- Yen, J.Y., 1971. Finding the K shortest loopless paths in a network. *Management Science* 17, 712–716.
- Ziliaskopoulos, A., Mahmassani, H., 1993. Time-dependent, shortest-path algorithm for real-time intelligent vehicle highway system applications. *Transportation Research Record* 1403, 94–100.