Trends in

Applied Sciences Research

ISSN 1819-3579

Academic
Journals Inc.

www.academicjournals.com

Trends in Applied Sciences Research 10 (5): 278-285, 2015
ISSN 1819-3579 / DOI: 10.3923/tasr.2015.278.285

K-Objective Time-Varying Shortest Path Problem with Zero Waiting Times at Vertices

G.H. Shirdel and H. Rezapour
Department of Mathematics, Faculty of Basic Sciences, University of Qom, Qom, Iran

Corresponding Author: G.H. Shirdel, Department of Mathematics, Faculty of Basic Sciences, University of Qom, Qom, Iran Tel: +98 25 32103022

ABSTRACT

This study considers a k-objective time-varying shortest path problem, which cannot be combined into a single overall objective. In this problem, the transit cost to traverse an arc is varying over time, which depend upon the departure time at the beginning vertex of the arc. An algorithm is presented for finding the efficient solutions of problem and its complexity of algorithm is analyzed. Finally, an illustrative example is also provided to clarify the problem.

Key words: Time-varying optimization, K-criteria shortest path problem

INTRODUCTION

Special form of the bi-criteria path problems was introduced by Hansen (1980). The number of pareto paths set were defined, where it grew exponentially with the number of nodes set. He solved the monotone bi-criteria and the bi-objective path problems by using a label setting algorithm. A multiple label setting algorithm was expanded to generate pareto shortest path by Martins (1984). Moreover, Corley and Moon (1985) applied the dynamic programming to solve the multi-criteria shortest path problem. Brumbaugh-Smith and Shier (1989) proposed the linear time algorithm to solve the bi-criteria shortest path problems. Several algorithm to solve bi-objective multi-model shortest paths by using bidirectional search were presented by Artigues *et al*. (2013), where path viability constraints are modeled by a finite state automaton. A comprehensive survey on multi-criteria shortest path algorithm is given by Ehrgott and Gandibleux (2002). The reader is referred to Ahuja *et al*. (1993), Artigues *et al*. (2013), Bertsekas (1991), Getachew *et al*. (2000), Reinhardt and Pisinger (2011) and Schrijver (2003) for developments in that area.

In time-dependent version of problem, Cooke and Halsey (1966) described the fastest path problem. Kostreva and Wiecek (1993) extended the research of Cooke and Halsey (1966) to the multi-criteria case. Getachew *et al*. (2000) developed the results of Kostreva and Wiecek (1993). Moreover, they replaced the non-decreasing arc costs constraints by bounds on the cost and relaxed the time grid constraints. Two-objective function with time dependent data was studied by Hamacher *et al*. (2006). The problem called the time-dependent bi-criteria shortest path problem. Moreover, they reviewed an algorithm proposed by Kostreva and Wiecek (1993), presented a new label setting algorithm and compared both algorithms numerically. Sha and Wong (2007) considered the best path with multi-criteria in time-varying network, where a transit time b (x, y, t) is needed to traverse an arc (x, y). They supposed the time-varying version of the minimum cost-reliability ratio path problem.

Consider a time-varying network flow G (V, A, b, c^r), r = 1, 2,..., k, where, V is the set of vertices with |V| = n, A is the set of arcs with |A| = m, b (i, j, t) is a transit time by departure time t at vertex i, where, (b, i, t) for each arc (i, j) and each time t are positive integers. Assume that each arc (i, j) is related with k features c^r (i, j, t), r = 0, 1,..., k, where, c^r (i, j, t), r = 0, 1,..., k are given arbitrary integers. These features may state costs or other transit factors. We suppose c^1 (i, j, t), c^2 (i, j, t),..., c^k (i, j, t) are k arbitrary integer transit costs by departure time t at vertex i for traveling from vertex i to vertex j on arc (i, j).

Consider the network contains no parallel arcs and loops. If the network contains parallel arcs or self-loops, we can convert it into one with no parallel arcs and self-loops easily. Network parameters as c and c^r are dependent on the time t = 0, 1,..., T, where, T is time horizon to travel from source vertex s to another vertex in network. Moreover, T is a positive integer. Let $P = (i_1 = s, i_2,..., i_l = \rho)$ be a time-varying path from the source vertex s to the target vertex ρ. The waiting at any vertex is not allowable. The approach is given in this study holds for the problem has arbitrary waiting times or bounded waiting times in a similar idea.

Let:

$$\zeta_r(P) = \sum_{(i,j)\in P} c^r(i,j,t),\ r = 1, 2, ..., k$$

Therefore ζ_r (P), r = 1, 2,..., k are functions of P in terms of k mentioned features. The aim is to find an optimal path to minimize the k mentioned features ζ_1 (P), ζ_2 (P),..., ζ_r (P). Therefore we want to have:

$$\min_{P\in\mathbb{F}}\left\{\zeta_1(P) = \sum_{(i,j)\in P} c^1(i,j,t), \zeta_2(P) = \sum_{(i,j)\in P} c^2(i,j,t), ..., \zeta_k(P) = \sum_{(i,j)\in P} c^k(i,j,t)\right\} \tag{1}$$

where, $\mathbb{F}$ is the feasible paths set on the subject of exist constraints. The problem has k objectives, then determining an optimal path such that all of the values of ζ_1 (P), ζ_2 (P),... and ζ_k (P) are minimum, may not be possible, so we need efficient solutions for the problem. Problem efficient solutions are introduced in next section.

This study considers time-varying shortest path problem with k objective functions, where, waiting times at vertices are not allowable. In section materials and methods, primary definitions and concepts are presented. This section surveys the k-criteria time-varying shortest path with zero waiting times, presents an algorithm for solving the problem and surveys its complexity. Section results presents an example for the mentioned problem.

MATERIALS AND METHODS

In this section, some definitions and operators are introduced, which we will use in algorithm for finding optimal solution in problem. Then, a theorem is proved for solving the problem. After that, an algorithm is presented corresponding to theorem.

Definition 1: Consider a time-varying path from i_1 to i_l is denoted by $P(i_1\text{-}i_2\text{-}\ldots\text{-}i_l)$. Let $\alpha(i_q)$ be arrival time of a vertex i_q on $P(i_1\text{-}i_2\text{-}\ldots\text{-}i_l)$ such that $\alpha(i_q) = t_q \geq 0$:

$$\alpha(i_q) = \alpha(i_{q-1}) + b(i_{q-1}, i_q, \tau(i_{q-1})) \text{ for } 2 \leq q \leq l \tag{2}$$

where, $w(i_{q-1})$ and $\tau(i_{q-1})$ are waiting time value and departure time of vertex i_{q-1} for $2 \le q \le l$ on $P(i_1\text{-}i_2\text{-}\dots\text{-}i_l)$, respectively and we have:

$$\tau(i_{q-1}) = \alpha(i_{q-1}) \text{ for } 2 \le q \le l$$

Meanwhile, we let $\alpha(s) = 0$ for source vertex s.

Definition 2: If $P(i_1\text{-}i_2\text{-}\dots\text{-}i_l)$ be a time-varying path from i_1 to i_l then:

- The time of time-varying path $P(i_1\text{-}i_2\text{-}\dots\text{-}i_l)$ is determined by $\alpha(i_l)\text{-}\alpha(i_1)$, if $i_1 = s$ (s is source vertex) then the time of time-varying path $P(i_1 = s\text{-}i_2\text{-}\dots\text{-}i_l)$ is $\alpha(i_l)$, particularly
- Time-varying path $P(i_1\text{-}i_2\text{-}\dots\text{-}i_l)$ has time at most t if $\alpha(i_l)\text{-}\alpha(i_1) \le t$ and has time exactly t if $\alpha(i_l)\text{-}\alpha(i_1) = t$
- Corresponding to the arc features $c^r(i, j, t)$, $r = 0, 1, \dots, k$, the costs of time-varying path $P(i_1\text{-}i_2\text{-}\dots\text{-}i_l)$ is defined as follow:

$$\begin{array}{l}
\zeta_1 (P) = \zeta_1 (i_k) = \zeta_1 (i_{k-1}) + c^1 (i_{k-1}, i_k, \tau(i_{k-1})) \\
\zeta_2 (P) = \zeta_2 (i_k) = \zeta_2 (i_{k-1}) + c^2 (i_{k-1}, i_k, \tau(i_{k-1})) \\
\quad . \qquad . \qquad . \qquad . \\
\quad . \qquad . \qquad . \qquad . \\
\quad . \qquad . \qquad . \qquad . \\
\zeta_k (P) = \zeta_k (i_k) = \zeta_k (i_{k-1}) + c^k (i_{k-1}, i_k, \tau(i_{k-1}))
\end{array}$$

Where:

$$\zeta_r (i_1) = 0, r = 0, 1, \dots, k$$

It is clear if waiting times are not allowable at any vertices, then:

$$\zeta_r(P) = \zeta_r(i_l) = \zeta_r(i_{l-1}) + c^r(i_{l-1}, i_l, \tau(i_{l-1})) = \sum_{(i,j)\in P} c^r(i, j, t), r = 1, 2, \dots, k \tag{3}$$

Definition 3: A time-varying path $P(i_1 = s\text{-}i_2\text{-}\dots\text{-}i_l = j)$ is more efficient than another $P'(i_1 = s\text{-}i_2\text{-}\dots\text{-}i_l = j)$ if $\zeta_1 (P) < \zeta_1 (P')$. If $\zeta_1 (P) = \zeta_1 (P')$ then $\zeta_1 (P) < \zeta_1 (P')$,..., also, if $\zeta_1 (P) = \zeta_1 (P')$, $\zeta_2 (P) = \zeta_2(P')$,..., $\zeta_{k-1} (P) = \zeta_{k-1} (P')$ then $\zeta_k (P) < \zeta_k (P')$.

Definition 4: A time-varying path $P(i_1 = s\text{-}i_2\text{-}\dots\text{-}i_l = j)$ is most efficient path in the feasible paths set $\mathbb{F}$, if there is not any path $P' \in \mathbb{F}$ such that P' is a more efficient than p.

Definition 5: Consider that $A_1 = \{(a_{11}, a_{12}, \dots, a_{1n}), (a_{21}, a_{22}, \dots, a_{2n}), \dots, (a_{k1}, a_{k2}, \dots, a_{kn})\}$ and $A_2 = \{(b_{11}, b_{12}, \dots, b_{1n}), (b_{21}, b_{22}, \dots, b_{2n}), \dots, (b_{k1}, b_{k2}, \dots, b_{kn})\}$ and A_1, A_2 are arranged in non-decreasing order lexicographically. The operator $\langle , \rangle$ on A_1 and A_2 is defined as follow:

$$\begin{array}{c}
A = \langle A_1, A_2 \rangle = \{(a_{11}, a_{12}, \dots, a_{1n}), (b_{11}, b_{12}, \dots, b_{1n}), (a_{21}, a_{22}, \dots, a_{2n}), \\
(b_{21}, b_{22}, \dots, b_{2n}), \dots, (a_{k1}, a_{k2}, \dots, a_{kn}), (b_{k1}, b_{k2}, \dots, b_{kn})\}
\end{array}$$

where, A is arranged in non-decreasing order lexicographically and keeps all elements in A_1, A_2.

Definition 6: Let $A_1 = \{(a_{11}, a_{12},..., a_{1n}), (a_{21}, a_{22},..., a_{2n}),..., (a_{k1}, a_{k2},..., a_{kn})\}$ and $(a_{01}, a_{02},..., a_{0n})$. Then:

- An add operator $\oplus$ is defined as follow:

$$A_1 \oplus (a_{01}, a_{02},\dots, a_{0n}) = (a_{11}, a_{12},\dots, a_{1n})+ (a_{01}, a_{02},\dots, a_{0n}),\dots, (a_{01}, a_{02},\dots, a_{0n})+ (a_{k1}, a_{k2},\dots, a_{kn})$$

  and for $A_1 = \varnothing$, we have: $A_1 \oplus (a_{01}, a_{02},..., a_{0n}) = (a_{01}, a_{02},..., a_{0n})$
- Eff $\{A_1\}$ is an operator that recognizes all of efficient elements of A_1 corresponding order lexicographically

Note that an operator eff{} can be sorted all of elements A_1 in non-decreasing order lexicographically by k comparison. In the following, a theorem is proved for solving the problem, first. Then an algorithm is presented corresponding to theorem. Let us introduce the following definition of ξ_x^t.

Definitions 7: Consider $P_1, P_2,..., P_w$ be w time-varying paths from source vertex s to vertex j with associated features $A_1 = (\zeta_1 (P_1), \zeta_2 (P_1),..., \zeta_k (P_1))$, $A_2 = (\zeta_1 (P_2), \zeta_2 (P_2),..., \zeta_k (P_2)),..., A_w = (\zeta_w (P_1), \zeta_w(P_1),..., \zeta_w(P_1))$, respectively. Let ξ_j^t be a efficient path from s to vertex j with time exactly t in which no waiting times are allowed at any vertices, i.e., $\xi_j^t = \text{eff}\{A_1, A_2,..., A_w\}$. If there is not any path from s to vertex j with time exactly t, let $\xi_j^t := \overline{\infty} = (\infty,\infty,...,\infty)$, moreover we define $\overline{0}$: = (0, 0,..., 0).

Theorem 1: $\xi_s^0 = \overline{0}$ and $\xi_j^0 = \overline{\infty}$ for $j \in V \backslash \{s\}$. For t>0 and $j \in V \backslash \{s\}$, we have:

$$\xi_j^t = \text{eff}\left\{ \bigcup_{\{i|(i,j)\in A\},\{u|u+b(i,j,u)=t\}} \left\{ \xi_i^u \oplus \left(c^1(i,j,u), c^2(i,j,u),...,c^k(i,j,u)\right)\right\}\right\} \tag{4}$$

Proof: By definition, it is simple to see that $\xi_s^0 = \overline{0}$ and $\xi_j^0 = \overline{\infty}$ for $j \in V \backslash \{s\}$. Second part of the theorem is proved by induction on t. For t = 1 the theorem obviously holds. Suppose the formula is correct for t' = 1, 2,..., t-1. Consider vertex j. If $\xi_j^t = \overline{\infty}$, then there is nothing to prove, else if ξ_j^t is infinite, assume $\xi_j^t = \xi_i^u \oplus \left(c^1(i, j, u), c^2(i, j, u),..., c^k(i, j, u)\right)$ for i and u such that $(i, j) \in A$ and u+b (i, j, u) = t, respectively. Notice that b (i, j, u)>0 then u<t, therefore, by induction there is a path P' from source vertex s to vertex i with time exactly u and efficient path ξ_i^u. The path P can be founded by extending P' to P by adding arc (i, j), so by u+b (i, j, u) = t, the time of P is exactly t. Now, we proved there exists a path with time exactly t and efficiency ξ_i^t. It is easy to show ξ_i^t is an efficient solution for problem. Consider that i and u are the predecessor of j on path P and the time of the subpath P' from s to i, respectively. Let ξ(P') be the cost of P'. Since, u+b (i, j, u) = t, then u<t, therefore, by induction, $\xi(P') \geq \xi_i^u$. By the definition, the cost of P is obtained as follows:

$$\xi(P) = \xi(P') \oplus \left(c^1(i, j, u), c^2(i, j, u),..., c^k(i,j,u)\right) \geq \xi_i^u \oplus \left(c^1(i, j, u), c^2(i, j, u),..., c^k(i, j, u)\right)$$

Again, according to the equation:

$$\xi_i^t = \xi_i^u \oplus \left(c^1(i, j, u), c^2(i, j, u),..., c^k(i, j, u)\right) \leq \xi(P)$$

Then we have must $\xi(P) = \xi_i^t$.

The following algorithm can now be presented by using the Theorem 1.

Algorithm efficient path in time-varying shortest path problem with k objectives

Begin

Initialize $\xi_s^0 = \{(\xi_1^0(s), \xi_2^0(s), ...\xi_k^0(s))\} := \{(0, 0,..., 0)\} := \bar{0}$

$\xi_s^t := \bar{\infty}$ for t>0 and $\xi_j^t := \bar{\infty}$ for $0 \le t \le T$ and $j \in V \setminus \{s\}$

Sort all values u+b(i, j,u) = t for u+0, 1, 2,..., T and for all arcs (i, j)∈A;

For t = 1, 2,...,T do

For each (i, j)∈A and each u such that u+b(i, j, u) = t, do:

$$\bar{\xi}_j^t := \xi_i^u \oplus \left(c^1(i, j, u), c^2(i, j, u),..., c^k(i, j, u)\right)$$

$$\xi_j^t := \left\langle \xi_j^t, \bar{\xi}_j^t \right\rangle$$

$$\xi_j^t := \text{eff}\left\{\xi_j^t\right\}$$

Let:

$$\xi^*(j) = \text{eff}\left\{\bigcup_{0 \le t \le T} \xi_j^t\right\}$$

End

Theorem 2: Above algorithm can be implemented in O(T^2mnC) time, where:

$$C = \max_{(i, j) \in A,\ t = 0,..., T} c^1(i, j, t)$$

Proof: The initialize step need O(Tn). In next step, to sort u+b(i, j,u) = t, there are mT values to be sorted, then this step needs O(Tm). Since, $\left|\bar{\xi}_j^t\right| \le nTC$, calculation of $\left|\bar{\xi}_j^t\right|$, merging $\bar{\xi}_j^t$ with ξ_j^t and the number of comparisons ξ_j^t can be done in O(nTC). Since, this step has to perform for all t = 0,1,...,T and all (i, j)∈A, it takes in total O(mTnTC) = O(T^2 mnC). Finally, the last step of computing ξ* for all j∈V, takes O(Tn) time. Therefore the overall time of algorithm is bounded by O(T^2 mnC).

RESULTS

In this section, a numerical example is examined to illustrate presented algorithm for finding efficient paths in time-varying shortest path problem, where each arc (i, j)∈A is associated with three attributes $c^r(i, j, u)$, r = 1, 2,3..

Consider a time-varying network as shown in Fig. 1, in which no waiting times are allowed at any vertices. Assume that the time horizon is T = 8 and we have:

- For arc (r, x) and for each t∈{0,1,...,8}, let: (b, c^1, c^2, c^3) = (1, 3, 1, 2)
- For arc (r, v) and for each t∈{0,1,...,8}, let: (b, c^1, c^2, c^3) = (1, 2, 1, 3)
- For arc (x, v) and for each ∈, let: (b, c^1, c^2, c^3) = (1, 3, 2, 4)

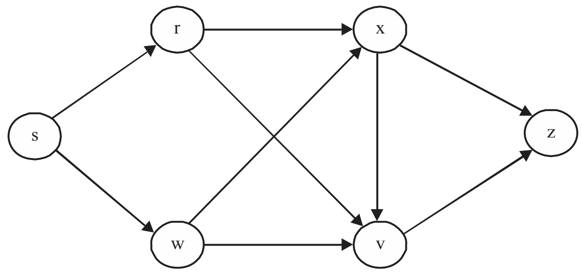

Fig. 1: A time-varying network for numerical example

Table 1: Transit times b and transit costs c^1, c^2, c^3 for Fig. 1

	Arcs (s, r)				Arcs (s, w)				Arcs (w, x)			
t	b	c^1	c^2	c^3	b	c^1	c^2	c^3	b	c^1	c^2	c^3
t = 0	1	3	2	2	1	4	2	4	2	2	1	4
t = 1	1	3	3	4	2	3	3	3	1	2	1	3
t = 2	2	4	2	1	1	4	5	1	1	2	3	3
t = 3	1	2	3	3	1	2	3	1	1	3	4	1
t = 4	2	4	3	2	1	2	2	1	2	1	2	4
t = 5	3	2	3	1	2	2	1	2	1	1	4	2
t = 6	1	3	1	3	1	3	2	1	3	2	3	4
t = 7	3	4	1	2	2	3	3	2	3	2	5	1
t = 8	2	3	1	2	1	3	2	3	1	5	2	7
	Arcs (w, v)				Arcs (x, z)				Arcs (v, z)			
t	b	c^1	c^2	c^3	b	c^1	c^2	c^3	b	c^1	c^2	c^3
t = 0	1	2	1	3	3	3	2	7	4	5	5	1
t = 1	3	3	1	2	2	2	3	4	1	2	7	1
t = 2	1	2	2	2	1	2	2	3	2	3	4	1
t = 3	1	1	1	1	1	3	2	2	2	4	3	2
t = 4	1	1	2	1	1	4	2	1	4	1	1	0
t = 5	1	2	1	1	2	1	2	1	1	6	2	3
t = 6	1	3	2	1	1	2	3	2	1	2	4	5
t = 7	2	4	2	1	1	3	2	5	2	2	2	1
t = 8	2	2	4	6	2	4	4	2	3	3	2	1

Table 2: Calculation of efficient paths for Fig. 1

j t	S	r	W	x	v	z
0	$\bar{0}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$
1	$\bar{\infty}$	3, 2, 1	4, 2, 4	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$
2	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	6, 3, 4	5, 3, 5	$\bar{\infty}$
3	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	9, 5, 8	8, 5, 7
4	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	7, 3, 6	8, 7, 6
5	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	13, 8, 10
6	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$
7	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$
8	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	$\bar{\infty}$	8, 4, 6
ξ*(j)	$\bar{0}$	3, 2, 1	4, 2, 4	6, 3, 4	5, 3, 5	8, 4, 6

The other transit times b (i, j, t) and transit costs (c^1, c^2, c^3) are listed in Table 1.

The results are shown in Table 2, by applying mentioned algorithm. When T = 8, the cost of efficient path connecting s to z is $\xi_z^{12} = (8, 4, 6)$ by path P(s-r-x-z), moreover this path is efficient path because we have:

$$\xi^*(z) = \text{eff}\left\{\bigcup_{0\le t\le 12} \xi_z^t\right\} = \xi_z^{12}$$

We can reach vertex r at t = 1, vertex w at t = 1, vertex x at t = 2, vertex v at t = 2, 3, 4 and vertex z at t = 3, 4,5, and t = 8, respecting Table 2. Thus, when T = 8, the cost of the shortest path connecting source vertex s to target vertex z with 3 objectives is (8, 4, 6), where 8, 4 and 6 are associated with c^1, c^2 and c^3, respectively.

By a back tracking procedure, the shortest path P* = (s-w-v-z) with 3 objectives is obtained, easily. Moreover, the arrival times of vertices on optimal efficient path P* = (s-w-v-z) are: $\alpha(s) = 0$, $\alpha(w) = 1$, $\alpha(V) = 4$ and $\alpha(Z) = 8$. Moreover, the time of P* is 8 and $\xi^*(P^*) = \xi^*(z) = (8, 4, 6)$.

The case of k-objective time-varying shortest path problem with respecting waiting times at Vertices, has been much less addressed. Getachew *et al*. (2000) considered the non-decreasing arc cost assumption by lower and upper bounds on the cost and relaxed the time grid assumption. Hamacher *et al*. (2006) were studied time-dependent bicriteria shortest path problems. They considered two objectives and assumed that travel time to be constant for the duration of travel along that arc, i.e., they considered the model of travel time is known as a frozen arc model. Artigues *et al*. (2013) proposed a bi objective shortest path problem in a multi-modal urban transportation network. They modeled their problem by a multi-layered network. Moreover, they considered both the " minimum time " and " minimum number of transfer " objectives.

This example considers a general model of time-varying network in which the transit times and transits costs are varying over time. Moreover, the k objectives are varying over time too. The problem is to find an optimal efficient path from source vertex to target vertex, so that, the total time of the path is at most T, where T is a given integer horizon time.

CONCLUSION

This study presented k-objective time-varying shortest path problem, where these objectives cannot combined into a single objective. An algorithm was proposed to handle the situation, where waiting times at vertices are zero and its complexity was analyzed. The waiting times were assumed be zero, therefore all waiting costs were considered equal to zero. This assumption can be relaxed without much difficulty and this algorithm holds for other situations in which waiting times at vertices are arbitrary or bounded.

REFERENCES

Ahuja, R.K., T.L. Magnanti and J.B. Orlin, 1993. Network Flows: Theory, Algorithms and Applications. Prentice Hall, Englewood Cliffs, N.J.

Artigues, C., M.J. Huguet, F. Gueye, F. Schettini and L. Dezou, 2013. State-based accelerations and bidirectional search for bi-objective multi-modal shortest paths. Transport. Res. Part C: Emerging Technol., 27: 233-259.

Bertsekas, D., 1991. Linear Network Optimization: Algorithms and Codes. MIT Press, Cambridge, MA., ISBN-13: 9780262023344, Pages: 359.

Brumbaugh-Smith, J. and D. Shier, 1989. An empirical investigation of some bicriterion shortest path algorithms. Eur. J. Opera. Res., 43: 216-224.

Cooke, K.L. and E. Halsley, 1966. The shortest route through a network with time-dependent intermodal transit times. J. Math. Anal. Appl., 14: 493-498.

Corley, H.W. and I.D. Moon, 1985. Shortest paths in networks with vector weights. J. Optim. Theory Applic., 46: 79-86.

Ehrgott, M. and X. Gandibleux, 2002. Multiple criteria optimization. State of the Art Annotated Bibliographic Surveys, Kluwer, Boston.

Getachew, T., M. Kostreva and L. Lancaster, 2000. A generalization of dynamic programming for Pareto optimization in dynamic networks. RAIRO Operat. Res., 34: 27-47.

Hamacher, H.W., S. Ruzika and S.A. Tjandra, 2006. Algorithms for time-dependent bicriteria shortest path problems. Discrete Optim., 3: 238-254.

Hansen, P., 1980. Bicriterion Path Problems. In: Multiple Criteria Decision Making: Theory and Applications, Fandel, G. and T. Gal (Eds.). Vol. 177, Springer, New York, USA., pp: 109-127.

Kostreva, M.M. and M.M. Wiecek, 1993. Time dependency in multiple objective dynamic programming. J. Math. Anal. Applic., 173: 289-307.

Martins, E.Q.V., 1984. On a multicriteria shortest path problem. Eur. J. Operat. Res., 16: 236-245.

Reinhardt, L.B. and D. Pisinger, 2011. Multi-objective and multi-constrained non-additive shortest path problems. Comput. Operat. Res., 38: 605-616.

Schrijver, A., 2003. Combinatorial Optimization: Polyhedra and Efficiency. Springer, New York, USA.

Sha, D. and C.K. Wong, 2007. Time-Varying Network Optimization. Springer, New York, USA., ISBN-13: 9780387712154, Pages: 264.