Dynamic vehicle routing in road evacuation: route design experimentation

A. Polimeni & A. Vitetta
Università degli Studi Mediterranea di Reggio Calabria,
DIMET - Dipartimento di Informatica, Matematica,
Elettronica e Trasporti, Italy

Abstract

In this paper, a shortest paths algorithm to find the paths in a time-dependent network is implemented and tested both on test and real networks. In the application on the real network, the data (obtained with real-time observations) are integrated with forecast data (to define the costs on non-observed network elements) obtained from a dynamic traffic simulation. The real case application is performed in emergency conditions; the aim is to optimize emergency vehicle routes (defined as a path succession) to reduce intervention times.
Keywords: time-dependent network, dynamic shortest path algorithm, route design, emergency conditions.

1 Introduction

In this paper we report an experiment related to paths and routes design for emergency vehicles in a time-dependent network. In a previous work [1] a model for path and route design in time-dependent networks was discussed. For simplicity sake, we refer to [1] for the full model formulation. Travel time forecasts and best vehicle routing are particularly important in emergency conditions.

In a road network in emergency conditions it is essential to provide system users' information in real time. To achieve this objective, we must know the evolution of link cost functions. Starting from the observed data, it is possible to calibrate the link cost functions, simulating the system in a time slice [1, 2]. This approach allows us to forecast link costs as time varies and hence to design paths (at the limit, instant by instant). The time-dependent link cost [3, 4] is

WIT Transactions on the Built Environment, Vol 117,
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/SAFE110341

related to time variation in demand characteristics [5–9], supply configuration [10, 11], and to demand-supply dynamic interaction [12, 13]. Knowledge of the system evolution allows action to be planned on the system itself [14–18].

Methods have been proposed in a static approach [19–21] to design paths/routes for emergency vehicles when a disaster occurs. In this paper, an update of such methods is proposed to take into account the characteristics of a time-dependent network.

The advances made by our study lie in the proposal of a new algorithm to find dynamic shortest paths and its application to a test network. In addition, the algorithm was applied on a real network for path and route design. The real network experiment was performed by considering emergency vehicles during an evacuation.

The paper is structured as follows. In section 2 the notation and definitions used in the paper are reported. In section 3 the proposed algorithm is discussed. Two applications are reported in section 4: the first on a test problem, the second on a real network. Finally, in section 5, some conclusions are reported.

2 Notation and definitions

We recall the notations used in [1]:
i, j, the initial and final nodes of the link (i, j);
v, a vehicle;
$A_{i,v}$, the instant vehicle v arrives at node i;
$\chi_{ij}(t)$, the cost function for the link (i,j) (the function defines the ij travel link cost for a vehicle v that arrives in node i at instant $A_{i,v}$);
$\omega_i(t)$, the waiting function at node i;
$\lambda_{ij}(t) = \chi_{ij}(t) + \omega_i(t)$;
$A^*_{i,v}$, the instant when $\lambda_{ij}(t)$ is minimum;
$w^*_{i,j,v} = A^*_{i,v} - A_{i,v}$;
$w^*_{i,j}$, the Optimum Waiting Time (OWT) at node i to reach j;
G(N, L), a graph with N sets of nodes and $L \subseteq N \times N$ the link set;
$C \subseteq N$, the set of centroid nodes, each centroid corresponding to the beginning and end of a path.

A path is a links sequence, without loops, that connects an initial node (origin) and a final node (destination). A vehicle route is a paths sequence which a vehicle uses to reach specific points (in this paper, users to rescue) on a road network. Hence, the path design consists in optimizing the link sequence; the route design consists in optimizing the paths sequence in a route.

In a time-dependent network a vehicle is subject to a cost that depends on time according to some functions [22, 24]. Note that the path cost is the sum between the link cost, for the links belonging to the path (similarly to a static network) and the nodes wait (if the wait is allowed). Consequently, the route cost (sum of the path cost, for the paths belonging to the path, and the operative costs) depends on the time.

www.witpress.com, ISSN 1743-3509 (on-line)

The path problem from an origin o can be formulated as:

$$z_j = argMin_{t,\, i \neq j} \, (A_{i,v} + \lambda_{ij}(t)) \tag{1}$$

where

z_j is the shortest path cost from origin *o* to node *i*;

The vehicle routing problem can be formulated as:

$$\zeta = argMin_{\xi} \, \Sigma_i \Sigma_j \Sigma_\kappa \, z_{i,j}(t) \cdot \xi_{i,j,\kappa} \tag{2}$$

subject to

capacity constraints (for each vehicle the capacity cannot be violated);
congruence constraints (a user cannot be reached more than once, all vehicles return to the starting point);
time-dependent constraints,

where

ζ is the solution cost;
$\xi_{i,j,\kappa}$ is a binary variable equal to 1 if the path between i and j (i, j $\in$ C); belongs to route κ, zero otherwise;
$z_{i,j}(\tau)$ is the shortest path cost between i and j, with i, j $\in$ C.

Equation (1) is the shortest path problem formulation, based on a time-generalized Bellman optimality condition ([1]). Equation (2) is the routes design formulation, based on optimum model ([1]).

In relation to the algorithm reported in section 3, the following notation is used:

FS(i), Forward Star of node i (i $\in$ N);
o, origin node (o $\in$ C);
z, node cost vector with component z_i is the cost to reach node i starting from the origin;
b, node list to visit;
e, node label vector with component e_i is the label to node i;
p, node predecessor vector with component p_i is the predecessor to node i.
All variables with subscript *o* are related to the origin.
It is assumed that the graph is connected, that is that for every pair of centroids (i, j; i $\in$ C, j $\in$ C) there is at least one path connecting i to j (which is also the shortest).

A shortest path tree, with its root in origin o, is a subgraph with only one route between each nodes pair.

3 Algorithm

To solve the path problem, various algorithms are proposed in literature [25, 26]. To solve the problem (1) a procedure is proposed that allows the shortest paths tree to be built for each origin o in a time-dependent network, optimizing iteratively the waiting time at the intermediate nodes. This procedure is based on Dijkstra algorithms, with some modification related to the problem dynamics.

In step 0, note the origin node *o*, the variables are initialized. The origin node *o* cost z_o is zero, the other node costs are infinity. Moreover, the predecessor vector is initialized. The node list **b** to visit is updated inserting the sequence (o, i, z_i), where $i \in FS(o)$. From the list the sequence (i, j, z_j) is extracted that has minimum cost z_j and is considered the forward star FS(j) from node j.

In step 1 for each node $u \in FS(j)$ the cost z_u is valued by optimizing the waiting time $w^*_{j,u}$ at node *j* (equivalently $A^*_{i,v}$ may be found), with $w^*_{j,u}$ defined in section 2. The time-generalized Bellman condition is evaluated: if the cost $z^*_u = (z_j + w^*_{j,u,v} + \chi_{ju}(A^*_{j,v}))$ at node u is less than their current cost z_u, then $z_u = z^*_u$ and the sequence (j, u, z_u) is inserted in the node list to visit. The node i label is updated and i is a definitive node; node i is the predecessor of nodes j: p_j=i.

In step 2 the procedure is stopped when or $\mathbf{b} = \varnothing$ or all destination nodes are definitive. Let an origin/destination pair (o/d) the path topology is determined starting from the destination and considering the predecessor node. Figure 1 shows the algorithm in question.

Step 0: initializing variables
{ o is the root of the shortest path tree; z_o=0; $z_i = \infty\ \forall\ i \in N$;
e_i=2 $\forall\ i \in N$; p_o= null; **b** =(o, i , z_i) $\forall\ i \in FS(o)$;}
Step 1: Bellman condition evaluation
{ extraction: (i, j ,z_j) with $z_j = \min_x(i, x, z_x)\ \ \forall\ (i, x, z_x) \in \mathbf{b}$;
$\forall\ u \in FS(j)$
if u is not a definitive node *then*
{ $A^*_{j,v} = \min_t \lambda_{ju} = \min_t \chi_{ij}(t) + \omega_i(t)$;
$w^*_{j,u,v} = A^*_{j,v} - A_{j,v}$; $z^*_u = (z_j + w^*_{j,u} + \chi_{ju}(A^*_{j,v}))$;
if $z^*_u < z_u$ *then*
{ $z_u = z^*_u$; (j, u ,z_u) is inserted in **b**;
e_i = 1; p_j=i;} remove (i, j ,z_j) from the list **b**;}}
Step 2: Test
if $\mathbf{b} = \varnothing$ or all destination nodes are definitive, *then* stop
else GoTo step 1

Figure 1: Proposed algorithm.

4 Applications

In this section two applications are reported: the first in a test network (section 4.1), the second in a real network (section 4.2).

4.1 Test network application

In this section we report some numerical applications of the proposed algorithm to a test network (figure 2). The objective is to test the proposed procedure before applying it to a real case. The test network consists of 6 nodes and 14 links; node 1 is the origin, and node 6 the destination.

We consider that all nodes have the same type of cost function in terms of specification. The three cases reported in [1] are termed (table 1):

1) linear: $c_{ij} = \chi_{ij}(t) = \alpha_1 + \alpha_2 \cdot t$;
2) exponential: $c_{ij} = \chi_{ij}(t) = \alpha_3 + \alpha_4 \cdot \exp(-\alpha_5 \cdot t)$;

3) periodic: $c_{ij} = \chi_{ij}(t) = \alpha_6 + \alpha_7 \cdot \sin(2\pi \cdot t/ \alpha_8 + \alpha_9)$;
with $\alpha_1 \ldots \alpha_9$ parameters to calibrate.

This also assumes that the waiting function is linear for each link.

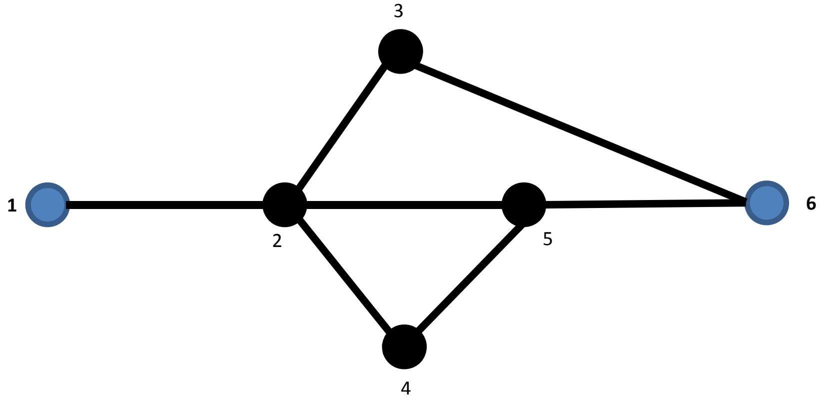


Figure 2: Test network.

Table 1: Test network link cost functions.

Link	Linear		Exponential			Periodic			
	α_1	α_2	α_3	α_4	α_5	α_6	α_7	α_8	α_9
1-2	10	0.00	1	1	0.200	10	1	1	0
2-1	30	-1.09	40	200	0.014	22	8	5	5
2-3	40	-1.00	20	100	0.014	45	9	5	0
2-4	20	-1.07	20	100	0.105	10	5	5	5
2-5	16	-1.08	40	200	0.002	10	5	5	10
3-2	20	-1.07	55	275	0.121	29	8	5	5
3-6	51	-1.18	50	250	0.103	31	5	5	5
4-2	40	-1.60	35	175	0.415	18	9	5	5
4-5	55	-1.05	40	200	0.001	12	5	5	0
5-2	28	-1.19	25	125	0.102	19	7	5	8
5-4	55	-1.07	20	100	0.101	17	5	5	5
5-6	50	-1.06	45	225	0.052	16	8	5	0
6-3	40	-1.09	30	150	0.102	14	7	5	10
6-5	50	-1.17	20	100	0.041	20	5	5	8

The experiment allows the shortest path to be found between 1 and 6, optimizing the waiting cost at some intermediate nodes. In table 2 the OWT at the node, conditioned by the node following is reported. For example, in the linear case node 5, in the case of a linear function, can have three values of OWT:
$w^* = 0$ if the node following is 2;
$w^* = 19.39$ if the node following is 4;
$w^* = 18.78$ if the node following is 6;

Table 2: Optimum waiting time for each case starting from node 1 at time zero.

Node	Link	w^* linear case	w^* exponential case	w^* periodic case
1	1-2	0.00	0.00	0.00
2	2-1	0.00	0.00	0.00
2	2-3	0.00	20.23	3.68
2	2-4	1.21	20.43	0.00
2	2-5	0.19	0.00	0.00
3	3-2	0.00	0.00	0.00
3	3-6	0.00	0.00	0.00
4	4-2	0.00	0.00	0.00
4	4-5	17.93	0.00	0.00
5	5-2	0.00	0.00	0.00
5	5-4	19.39	0.00	0.00
5	5-6	18.78	0.00	1.39
6	6-3	0.00	0.00	0.00

A similar consideration can be made for exponential and periodic cases. In table 3 the paths between origin and destination varying the cost functions are reported. We may note a reduction in costs compared to the path where the wait at nodes is not allowed.

Table 3: Path topology and cost.

Model	Topology	Travel cost with optimum wait			Travel cost without optimum wait			Cost reduction (%)
		Travel cost	Wait cost	Total cost	Travel cost	Wait cost	Total cost	
Linear	1-2-5-6	29.18	18.78	47.96	49.09	0	49.09	2.35
Exponential	1-2-3-6	146.08	20.23	166.31	169.24	0	169.24	1.76
Periodic	1-2-5-6	25.32	1.39	26.71	35.46	0	35.46	32.76

4.2 Real network application

The objective of the application is to test the proposed procedure in a real context in order to design the paths and routes for emergency vehicles in an evacuation scenario. With this procedure, taking into account the network dynamics, paths and routes can be designed by adapting them to network conditions that change over time.

The network consists of all the main roads of Melito di Porto Salvo (Italy) and is formed by 38 nodes and 66 links (figure 3). Some links (represented in figure 3 with a broken line) are reserved for emergency vehicles. In this application, six centroids are considered: one safe area (indicated by R in figure 3) is the point where users are to be led; five locations (indicated by A/E in figure 3) where weak users are located.

The work is based on an evacuation simulation in a town, where the paths chosen by an ambulance driver were monitored using an on-board GPS and video-cameras deployed on the network [27].

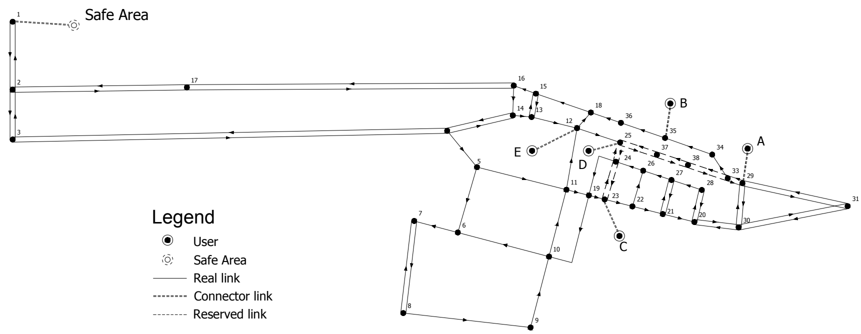
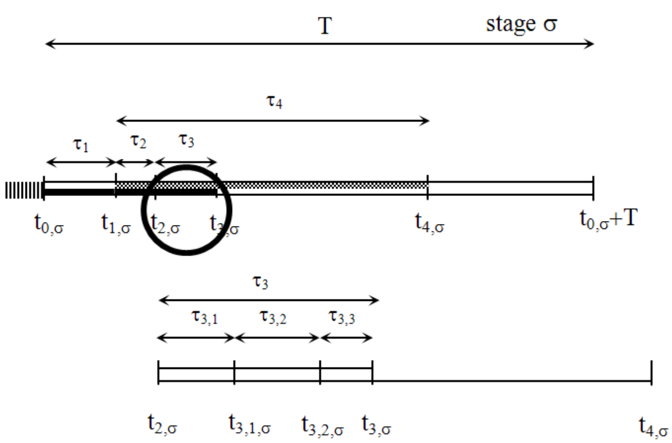


Figure 3: Melito Porto Salvo road network, localization of users and safe area.

A general framework to define analytic functions for within-day variations of travel time forecasts suitable for emergency vehicle route design is reported in [2]. The general framework integrates some modules working in time according to the concept of rolling horizon, schematically depicted in figure 4.



σ: stage
T: stage length
$t_{0,\sigma}$: start procedure
$[t_{0,\sigma}, t_{1,\sigma})$: assignment procedure
$(t_{1,\sigma}, t_{2,\sigma}]$: continuous travel time function calibration
$(t_{2,\sigma}, t_{3,1,\sigma}]$: building routes
$(t_{3,1,\sigma}, t_{3,2,\sigma}]$: computing routes cost
$(t_{3,2,\sigma}, t_{3,\sigma}]$: testing routes

Figure 4: The rolling horizon (adapted from [2]).

Considering a stage σ of length T (it is assumed that the forecasted costs are reliable in stage σ), the procedure starts at instant $t_{0,\sigma}$. In the interval $[t_{0,\sigma}, t_{1,\sigma})$ a traffic assignment allows link flows and travel times to be obtained which are used, in the interval $(t_{1,\sigma}, t_{2,\sigma}]$, for calibrating a continuous cost function [2]. In the interval $(t_{2,\sigma}, t_{3,\sigma}]$ the route design problem is solved. This interval is separated into three sub-intervals, to highlight some aspects of route design. In the sub-interval $(t_{2,\sigma}, t_{3,1,\sigma}]$ the travel time functions from traffic assignment and other real information on the system state are received (i.e. the users to rescue and vehicle locations). In the sub-interval $(t_{3,1,\sigma}, t_{3,2,\sigma}]$ the routes are designed with the procedure reported in section 4.2.2. Finally, in the sub-interval $(t_{3,2,\sigma}, t_{3,\sigma}]$ the new information is communicated to the vehicle.

The shortest path search and route design are two joint problems, but for simplicity sake they are presented separately in the next subsections.

4.2.1 Shortest path

We assume that the waiting function is linear and that the wait is allowed at each node. The aim is to optimize the waiting cost at some (at the limit, at all) intermediate nodes for reducing the path cost. In this example, we suppose that node R is the origin and then we find the shortest path (starting from the origin at time t = 0, the procedure is identical starting from the origin at another time) from the origin to all destinations. Similar considerations can be made by choosing any other origin. The cost function considered is a continuous and periodic function calibrated and validated in [2].

We observe that by optimizing the wait time at nodes, the path cost decreases for all paths considered. Table 4 reports the path cost when waiting is allowed and when waiting is not allowed.

Table 4: Path cost comparison from origin R.

Destination	Cost (waiting allowed)	Cost (waiting not allowed)	Cost reduction
A	173.00	173.97	0.56
B	194.62	195.27	0.33
C	133.41	134.29	0.66
D	159.57	160.14	0.36
E	139.50	140.27	0.55

4.2.2 Vehicle routing

The observed vehicle reaches five users; for each stop the operation time is measured and known. The routes are constructed as follows. The first node is the depot (node 1). By applying the shortest paths search algorithm the paths from the safe area to all possible destinations are found. We assume that the node to insert is the first node and that:

- the insertion respects the problem constraints;
- the node is the closest to the previous in the route.

If the insertion in the route of a selected node violates one or more constraints, then the list scrolls to find the *second* closest node to the previous node in the list: this step is repeated until a node is inserted or all nodes have been examined and none could be added. With this procedure, route topology and route cost are evaluated jointly.

We assume that the departure time from the safe area is t = 0 (that is the departure time from the safe area is the origin of the time axis). The departure time from a generic centroid (we assume that it is the sum between the arrival time and the operation time) defines the instant when the vehicles start from the centroid (and hence, the instant at which the paths to the not visited centroids are evaluated).

Along the designed route (table 5) for each centroid the arrival time (instant when the vehicle reaches the node) and the operation time (time needed for download operations) are reported. Two cases are considered: when waiting is allowed and when it is not. In the latter case, it can be observed that the route cost is reduced. The reduction is not significant in percentage terms. The objective of this paper is not to determine the percentage reduction. The objective is to test the possibility of applying dynamic vehicle routing in order to reduce the total time. If the travel time has greater variation in clock time, the improvement in percentage terms is significant.

Table 5: Route design.

Waiting allowed				**Waiting not allowed**			
Centroid	**AT[s]***	**OT[s]***	**RC[s]***	**Centroid**	**AT[s]***	**OT[s]***	**RC[s]***
R	0	0	0	R	0	0	0
C	133.41	389	522.41	C	134.29	389	523.29
A	561.81	207	768.81	A	562.61	207	769.61
B	789.65	480	1269.65	B	790.55	480	1270.55
E	1328.55	287	1615.55	E	1329.54	287	1616.54
D	1636.04	272	1908.04	D	1637.21	272	1909.21
R	2119.03	0	2119.03	R	2120.23	0	2120.23

*AT: Arrival Time, OT: Operation Time, RC: Route Cost

5 Conclusions

This paper reported a procedure to find shortest paths and an application on the routes design problem in a time-dependent network. Starting from a general theory developed in a previous work [1], the structure of an algorithm to find the shortest path in a time-dependent network was proposed. This algorithm, based on a time-generalized Bellman optimality condition, is an exact procedure that allows the path to be optimized by taking account of the wait at intermediate nodes.

Of the two applications reported, the first concerned a test network to test the procedure before applying it to a real network; the second concerned a real network where paths and routes were designed from a continuous and periodic function calibrated with observed data. In the real network application, a comparison was made between paths and routes where waiting is allowed or those where it is not allowed. It was found that waiting can reduce the route cost. Future developments will entail the design of a genetic algorithm to build the routes and specify and calibrate other cost functions, as well as test the models and procedure also in the context of routing for freight distribution.

References

[1] Polimeni A. & Vitetta A., Dynamic vehicle routing in road evacuation: an exact model for route design. Proc. of Urban Transport XVII, Urban Transport and the Environment in the 21st century, Brebbia C. A. (ed.), WIT Press, Southampton, 2011.

[2] Musolino G. & Vitetta A., Within-day traffic assignment and dynamic vehicle routing: travel time short-term forecasting in evacuation conditions. Urban Transport and the Environment in the 21st century, Brebbia C. A. (ed.), WIT Press, Southampton, 2011.

[3] Cooke K.L. & Halsey E., The shortest route through a network with time-dependent internodal transit times. Journal of Mathematical Analysis and Applications, 14(3), pp. 493-498, 1966.

[4] Dreyfus S.E., An appraisal of some shortest-path algorithms, Operations Research, 17(3), pp. 395-412, 1969.

[5] Russo F. & Chilà G., Safety of users in road evacuation: demand models. *Proc. of Urban Transport XIII, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 773-782, 2007.

[6] Russo F. & Chilà G., Safety of users in road evacuation: RP vs. SP surveys in demand analysis. *Proc. of Urban Transport XIV, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 703-713, 2008.

[7] Russo F. & Chilà G., Safety of users in road evacuation: Modelling and DSS for demand. *Proc. of Sustainable Development and Planning IV, vol.1*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 465-474, 2009.

[8] Russo F. & Chilà G., Dynamic approaches to demand model in evacuation conditions. *Proc. of Urban Transport XVI, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 303-312, 2010.

[9] Russo F. & Chilà G., A sequential dynamic choice model to simulate demand in evacuation conditions. *Proc. of Risk Analysis VII & Brownfields V*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 431-442, 2010.

[10] Russo F, Vitetta A., An assignment model with modified Logit, which obviates enumeration and overlapping problems. *Transportation*, 30 (2), pp. 177-201, 2003.

[11] Cantarella G. E., G. Pavone and A. Vitetta, (2006). Heuristics for urban road network design: lane layout and signal settings. European Journal of Operational Research, 175, pp. 1682–1695.

[12] Vitetta A., Musolino G. & Marcianò F.A., Safety of users in road evacuation: Supply and demand-supply interaction models for users. *Proc. of Urban Transport XIII, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 783-792, 2007.

[13] Vitetta A., Musolino G. & Marcianò F.A., Safety of users in road evacuation: calibration of cost functions and simulation. *Proc. of Urban Transport XIV, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 715-725, 2008.

[14] Russo F. & Rindone C., Safety of users in road evacuation: planning internal processes and guidelines. *Proc. of Urban Transport XIII, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 825-834, 2007.

[15] Russo F. & Rindone C., Safety of users in road evacuation: the logical framework approach in evacuation planning. *Proc. of Urban Transport XIV, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 751-760, 2008.

[16] Russo F. & Rindone C., Safety of users in road evacuation: Modelling and DSS for LFA in the planning process. *Proc. of Sustainable Development and Planning IV, vol.1*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 453-464, 2009.

[17] Russo F. & Rindone C., Evaluation methods for evacuation planning . *Proc. of Urban Transport XVI, Urban Transport and the Environment in the 21st century*, Brebbia C. A. (ed.), WIT Press, Southampton, pp. 335-343, 2010.

[18] Russo, F. & Rindone C., Data Envelopment Analysis (DEA) for evacuation planning. *Proc. of Risk Analysis VII & Brownfields V*, Brebbia C.A. (ed.), WIT Press, Southampton, pp.455-467, 2010.

[19] Vitetta A., Quattrone A. & Polimeni A., Safety of users in road evacuation: algorithms for path design of emergency vehicles, *Proc. of Urban Transport XIV, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 727-737, 2008.

[20] Yang, S., Hamedi, M. & Hagani, A., Online Dispatching and Routing Model for Emergency Vehicles with Area Coverage Constraints, *Transportation Research Record: Journal of the Transportation Research Board*,1923, 2005.

[21] Gong, Q., Jotshi, A., Batta, R. Dispatching/Routing of emergency vehicles in a disaster environment using data fusion concept, *Socio-Economic Planning Sciences*, 43, pp.1-24, 2009.

[22] Hall R.W., The fastest path through a network with random time-dependent travel times. Transportation Science, 20 (3), pp.182-188, 1986.

[23] Pattanamekar P., Park D., Rilett L.R., Lee J. & Lee C., Dynamic and stochastic shortest path in transportation networks with two components of travel time uncertainty. Transportation Research Part C, 11, pp. 331-354, 2003.

[24] Ahuja R.K., Orlin J.B., Pallottino S. & Scutellà M.G., Dynamic shortest paths minimizing travel times and costs. Networks, 4, pp.197-205, 2003.

[25] Dean B.C., Continuous-time shortest path algorithm. Massachusetts Institute of Technology, MSc thesis, 1999.

[26] Chabini I., Discrete dynamic shortest path problems in transportation applications: complexity and algorithms with optimal run time.

Transportation Research Record: Journal of Transportation Research Board 1645, pp. 170-175, 1998.

[27] Vitetta A., Quattrone A. & Polimeni A., Safety of users in road evacuation: design of path choice models for emergency vehicles. *Proc. of Urban Transport XIII, Urban Transport and the Environment in the 21st century*, Brebbia C.A. (ed.), WIT Press, Southampton, pp. 803-812, 2007.