

# A real-time information system for public transport in case of delays and service disruptions

Maurizio Bruglieri<sup>a</sup>, Francesco Bruschi<sup>b</sup>, Alberto Colomi<sup>a,c</sup>,  
Alessandro Luè<sup>a,c,\*</sup>, Roberto Nocerino<sup>c</sup>, Vincenzo Rana<sup>b</sup>

<sup>a</sup>*Dipartimento di Design - Politecnico di Milano, Via G. Durando 38/A, 20158 Milano, Italy*

<sup>b</sup>*Dipartimento DEIB - Politecnico di Milano, via Ponzio 34/5, 20133 Milano, Italy*

<sup>c</sup>*Consorzio Poliedra - Politecnico di Milano, via G. Colombo 40, 20133 Milano, Italy*

\* Corresponding author. Tel.: +39-02-2399-2905; fax: +39-02-2399-2911.

E-mail address: alessandro.lue@polimi.it

## 1. Introduction

Issues and problems of mobility services can have a deep and spread impact on other dimensions of the urban context. Far from being exhaustive, it is possible to mention: emission of pollutant and greenhouses gases, road congestion, accidents, and energy consumption, which all have severe effects on the environment (both at local and global scale) and on the quality of urban life. Promoting the use of public transportation and Intelligent Transport Systems may help in decreasing traffic congestion and air pollution in urban areas, and improving accessibility for all citizens may help to reduce these problems (Ciccarelli et al., 2006; Chorus et al., 2006; Grotenhuis et al., 2007; Luè and Colomi, 2015). The positive effects of ITS on reducing such impacts are well studied and established (Maccubbin et al., 2008; Naniopoulos et al., 2004; Kolosz and Grant-Muller, 2015): providing reliable and real-time updated information to the users, travel planners are one of the most effective ITS system for reducing the impact of mobility.

MOTUS (MObility and Tourism in Urban Scenarios) is a project financed by the Ministry of Economic Development. The general goals of the projects can be summarized as follows:

- Create a system, available to mobility managers and operators, to comprehend, characterize, analyze and monitor urban and extra-urban mobility, both of citizens and tourists.
- Create a system able to integrate data, collected by heterogeneous sources, for instance from existing infrastructures on the territory, e.g. Caccia Dominioni et al. (2008) and data from mobile devices, e.g. Bar-Gera (2007).
- Provide users with information services, making them able to manage their personal mobility and to choose the best and most sustainable mobility solution. In particular we want to yield to the users a decision tool useful especially in case of emergency or disruption of the public transport.

The paper describes one of the adopted solutions for addressing the third goal of MOTUS: the travel planner implemented has been specifically designed for public transport. Promoting the use of public transportation, in fact, may help in decreasing traffic congestion and air pollution in urban areas and improving accessibility for all citizens, including those who cannot use a private vehicle. The direct objective of MOTUS Travel Planner is to enable citizens and tourists to organize their trips with the Local Public Transport (LPT), even in case of emergency or disruption of the service.

The two main characteristics that make the MOTUS Travel Planner innovative and different from other travel planners already on the market are:

- It uses real time data about public transport, directly provided by the LPT manager in Milano, ATM (Azienda Trasporti Milanesi) through the E015<sup>†</sup> integration platform. The idea behind MOTUS is that the effectiveness and competitiveness of public transportation can be improved through an integrated system based on real-time data management. In general, poor information to customers is one of the major issues in public transportation services, which is the reason for allocating substantial efforts to implement a powerful information tool.
- It makes possible managing emergencies or disruptions that can affect the LPT network, simply putting offline one (or more than one) station of the LPT in an easy-to-use interface. This functionality is more and more useful in congested cities and it is very effective in case of big events (like fairs, exhibition, sport match, etc.).

The problem of finding the fastest journey in a public transportation according to the planned timetable has been widely studied in the literature. Most approaches model the transportation network as a graph where the problem is reduced to the computation of a shortest path (Müller-Hannemann, 2007). Several experimental studies show that such approach, with some suitable refinement, can also be used in practice (Bauer et al., 2011; Delling et al., 2009; Jariyasunant et al., 2010; Pyrga et al., 2014; Schulz, 2005).

<sup>†</sup> <http://www.e015.expo2015.org/>

Compared with standard road network, e.g. as for the algorithm developed in Bruglieri et al. (2014), urban timetabled public transport network is more difficult to model mainly because it is intrinsically time varying since certain arcs can be travelled only at specific times. Therefore the first problem consists in suitable modelling the timetable to allow the path computation. For this purpose two kinds of models are considered in the literature: the *time-expanded* model and the *time-dependent* one. The *time-expanded* model considers the departure of each mean from a stop as a time-varying event which happens in discrete instant times and builds in this way a *space-time graph* (often also called *event graph*) that “unroll” the time (Pallottino and Scutellà, 1998). The main disadvantage of such approach is that the resulting graph is often too large. Instead, the *time-dependent* model, generates graphs significantly smaller because the time is not “unrolled” since the time dependency is represented through a travel time function associated with the arcs, mapping departure times into travel times. In this contest, Muller-Hannemann et al. (2007) and Pyrga et al. (2008) developed models involving both time dependent and time-expanded mode for timetable information in public transport systems.

An algorithm that has recently found a wide field of application in practice has been that based on the so called *transfer patterns*, proposed by Bast et al. (2010). Such algorithm is used for instance by Google Maps. The main observation on the basis of such a method is that for two given stops, one can find and encode each sequence of intermediate transfer stations (i.e., stations where we change from one line to another) that can lead to an optimal route. The set of these sequences of transfers is called *transfer pattern*. These patterns can be precomputed, leading to very fast query times.

Recent methods avoid the construction of a graph processing the timetable directly (Delling et al., 2012; Delling et al., 2013). In particular, Delling et al. (2013)’s approach is focused on transportation lines and is useful to find all Pareto-optimal journeys considering as criteria the arrival time and the number of stops.

Concerning the reliability of journeys, Böhmová et al., (2013) address the problem of robust routing in urban public transportation networks. They propose solutions that are robust for typical delays, considering past observations of real traffic situations. In particular, they assume to have “daily records” containing the observed travel times in the whole network for a few past days.

To the best of our knowledge no paper addresses the study of a trip planner specifically built to manage disruptions in a public transportation network, despite the practical importance of such a situation. Therefore our work represents the first study that faces such a problem. The original contributions are a particular time-expanded representation of the public transport network on which the problem can be solved using a modified version of Dijkstra’s algorithm. Such representation makes very easy to model a wide range of service disruptions, much more meaningful than those possible with approaches currently proposed by transit agencies. For instance it is possible selectively inhibiting the getting on and/or off of a single ride, cancelling specific rides or modelling temporary deviations. The solution approach has been tested on the Milano city considering the General Transit Feed Specification (GTFS) data of public transport made available by the Agency for Mobility, Environment and Territory of the Municipality of Milano (AMAT). The model, at the moment, does not manage multimodal trips, i.e. it does not consider alternative transport means (e.g., cycling, car sharing), even though we recognize the importance of integrating, for instance, different kinds of sharing services (Arena et al., 2015; Luè et al., 2012), and thus consider it as a possible future development.

The rest of the paper is organized in the following way. In Section 2 we describe the solution approach developed and the features of the database used. In Section 3 we show some test results. Finally in Section 4 we draw some conclusions and we indicate the future works.

## 2. The MOTUS travel planner

### 2.1 Input data and exchange protocols

The data used by our travel planner are the open data of Milano public transport made available by AMAT. Such data are in *General Transit Feed Specification* (GTFS)<sup>‡</sup> format. Nowadays, GTFS has become a worldwide standard

<sup>‡</sup><https://developers.google.com/transit/gtfs>

format for the information on public transport concerning both their schedules and the geographical information on their stops. Indeed such a format allows public transport managers to publish through the so called *feeds*, updated information in a shared format. Each feed is made up of several text files describing in detailed way information like the stops associated with each line, the routes, the timetables and so on. Such data are then moved through a suitable driver in a *PostgreSQL Database* where they can be accessed and queried by the MOTUS platform. The GTFS feeds downloaded from the AMAT website<sup>§</sup> are imported into equivalent tables:

- The *Stop time table* includes all the information on the arrival time of each transport mean to each stop.
- The *Calendar table* lists, for each journey, the days for which the service is available.
- The *Route table* lists the Milano public transport lines and detailed information for each of them.
- The *Trip table* lists the journeys planned for each line.
- The *Stop table* lists all the stations where the Milano public transport stop.

Imported data are then processed to build a *time expanded graph* of the Milano public transport network.

## 2.2 The time-expanded model of the public transport network

The public transport network is modelled by means of a particular time-expanded graph characterized by nodes of two kinds: *i) station nodes*, representing physical stops; *ii) stop nodes*, representing the stops of each ride at specific station and time point for get-on/get-off. Hence, multiple stop nodes are associated with each station node. Each stop node, with the exception of the endpoints, is connected with the corresponding station by two arcs (get-on/get-off). A cost, representing the time for getting on/off the public transport mean, is associated with such arcs. Each stop node is linked with the subsequent one of the same journey by an arc ending in the latter and with zero cost. Each stop node is labelled with a *timestamp*, i.e., a value representing the arrival time at that node. Therefore the travel time for going from a stop to another one is not indicated on any arc, since such information is already embedded into the timestamp of each stop node. For instance, given a single line journey with four stops as depicted in Fig. 1, the corresponding time-expanded graph for three consecutive journeys starting at 12:00, 12:10 and 12:20, respectively, is represented in Fig. 2. In such a graph  $s_1, s_2, s_3$  and  $s_4$  represent the station nodes, while, for  $i=1, \dots, 4$ ,  $i', i'', i'''$  represent the corresponding stop nodes for each of the three journeys, respectively.

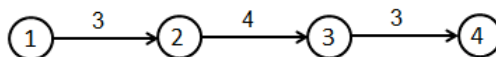


Fig. 1: Example of a single line journey

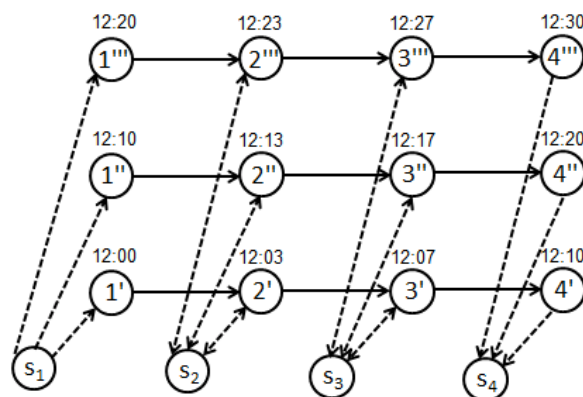


Fig. 2: Representation of the example depicted in Fig. 1 through the time-expanded graph

<sup>§</sup> <http://www.amat-mi.it/it/mobilita/dati-strumenti-tecnologie/dati-gtfs/>

Through such a time-expanded graph several types of disruptions can be easily modeled. The case where a station is closed for both the users' getting on/off and for the transit of the transport means can be represented straightforwardly eliminating in the time-expanded graph both the station node and the stop nodes corresponding to such a station, besides all incident arcs, as shown in Fig. 3. Instead, if a station is inhibited for the getting on/off of the users but not for the transit of the transport means, then only the station nodes are removed from the time-expanded graph (besides all incident arcs), as shown in Fig. 4. Finally, if a ride is canceled then all stop nodes involved in such a ride are removed from the graph together with all incident arcs, as shown in Fig. 5.

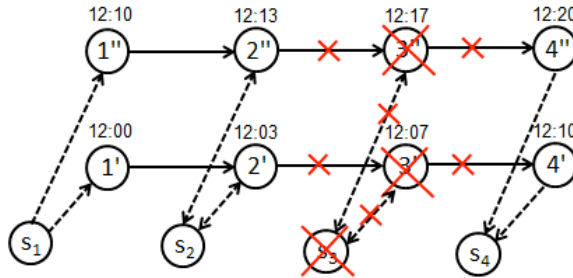


Fig. 3: Case of a disruption where a station ( $s_3$ ) is closed for both the getting on/off and the transit

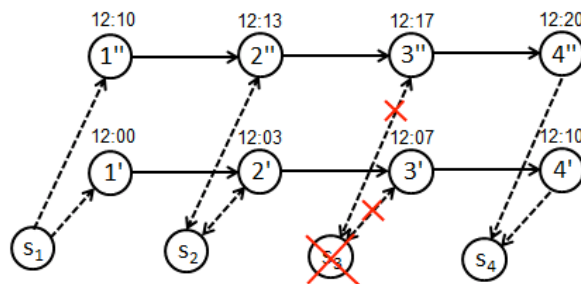


Fig. 4: Case of a disruption where a station ( $s_3$ ) is closed only for the getting on/off but not for the transit

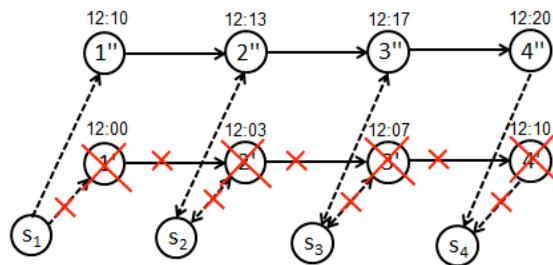


Fig. 5: Case of a disruption where a ride is canceled

### 2.3 Geocoding

To determine the GPS coordinates of the origin and of the destination of the user from his/her departure and arrival addresses, a geocoding software has been developed. Such a software is based on an open and collaborative map system. All the streets, civic numbers and interest points of Milano are collected by a map system and stored in a database together with the GPS coordinates. This database is then used by the MOTUS platform to suggest, for

instance, the street names to the users and convert them into coordinates. The map system adopted is OpenStreetMap (OSM) \*\*, a worldwide collaborative project whose main purpose is map creation. The main feature of OSM is its free license, the so called Open Database License, which allows to use the map with no constraint except from citing the source itself. Another important feature of OSM is that everybody can contribute to enrich the database or to correct the data, thus favoring the accuracy of the maps and their constant updating.

#### 2.4 The path computation algorithm

Through the geocoding software, the departure address and that of destination are converted into GPS coordinates. Then, a set of possible departure station nodes  $N_D$  and a set of potential arrival station nodes  $N_A$  are detected considering those contained in a circle centered in the GPS points associated with the departure and arrival address, respectively. The radius of such circle can be configured according to the maximum distance that a user is willing to walk. A node representing the user origin and one representing the destination are added to the time-expanded graph and are linked with all nodes of  $N_D$  and  $N_A$ , respectively.

On such a graph the following algorithm to compute the minimum arrival time path is applied, a modified version of the well-known Dijkstra's algorithm. Like in the latter, we associate with each node a label whose value can change during the iterations (representing the value of the best path found to reach the node until that iteration) and at end of the algorithm will represent the minimum arrival time to reach the node.

At the beginning, the origin node is assigned a label equal to departure time of the user. All stop nodes with timestamp lower than the departure time of the user are neglected since they correspond to events already happened i.e., rides already carried out. At each iteration, the node with minimum label (let it  $v$ ) is selected, its label becomes definitive and the labels of all its successors are updated. If the successor is a station node (let it  $w$ ), its label ( $L[w]$ ) is updated according to the usual rule of Dijkstra's algorithm:

$$L[w] = \min \{L[w], L[v] + c_{vw}\}$$

where  $c_{vw}$  is the cost of arc  $(v,w)$  and  $L[v]$  is the label of node  $v$ .

Differently from Dijkstra's algorithm, the label of each stop node is not computed through the previous formula, since it coincides with its timestamp. The algorithm ends when the label of one station node in  $N_A$  becomes definitive.

### 3. Testing the MOTUS travel planner

In order to highlight the innovative features of the MOTUS travel planner, we set up a comparison with other main travel planners, in presence of significant, crucial disruptions of some transit nodes. We then compared the paths proposed by the different engines with respect to their main characteristics (travel time, number of changes, walking distances, accuracy of time estimation etc.). The test aims at assessing the behavior of the MOTUS planner in normal conditions. To do this, we considered a set of five crucial transit points of Milan: Duomo, Rho Fiera, Cascina Gobba, Bovisa, Porta Romana. We then queried all three engines for the most significant  $O \rightarrow D$  couples. We did not constrain the planning engines in any way (all the means of transportation allowed, no limits on walking distance and/or changes): they had to just look for the fastest solution.

\*\* <http://www.openstreetmap.org>

Table 1. Comparison between Moovit, Google Transit and MOTUS travel planner  
(NAP = Number of Alternative Paths, TT = Travel Times in minutes, % = percentage of shared sub-path)

	MOOVIT				Google Transit				MOTUS	
	NAP	Paths	TT	%	NAP	Paths	TT	%	Paths	TT
Duomo → Rho Fiera	3	M1	30	0	1	M1	23	-	M1	28
		M3+S5	35							
		528+S6	88							
Duomo → Cascina Gobba	3	M1+M2	28	45	2	M1+M2	27	55	M3+M2	27
		M1+44	30			M3+M2	25			
		M1+86	41							
Duomo → Porta Genova	3	M1+M2	16	0	3	M1+M2	15	0	2	12
		2	20			2	16			
		14	19			M2	19			
Rho Fiera → Cascina Gobba	3	S6+M2	35	45	2	M1+M2	57	55	M1+M2	47
		S5+M2	35			M1+44	76			
		M1+M2	62							
Rho Fiera → Porta Genova	3	S6+M2	29	20	2	M1+M2	40	80	M1+M2	29
		S5+M2	29			M1+19	55			
		M1+M2	41							
Cascina Gobba → Porta Genova	3	M2	43	0	1	M2	26	-	M2	29
		44+M1+19	63							
		86+M1+19	66							

The data collected suggest some observations:

- 1) the path suggested by the three engines are mostly similar, with some exceptions. This comes as no surprise, since all three try to minimize travel time, and there are well known state-of-art algorithms that solve the problem in acceptable time. Travel times (TT) estimated by the three engines have different values also for the same paths (e.g., the trip “Duomo → Rho Fiera” with the M1 underground line), because they make different hypotheses on certain parameters (e.g., travel time by foot).
- 2) Google and Moovit propose alternative (more than one) paths most of the times. Typically, these paths share a significant portion of the travel. This implies that, even though the plurality of alternative paths offers some robustness to disruptions, they share possible common points of failure.

MOTUS, on the other hand, in normal condition proposes the single best path and, in case of need, accepts explicit information on different kinds of transit disruptions (unreachable stations, link unavailability), producing, in real time, optimal alternatives that bypass the disruption. The required CPU time is of few tenths of seconds.

Moreover, MOTUS planner can produce reroutes that are sensitive to the “travel status” of the passenger: if he is aboard a given means of transportation, the engine takes this information into account, which makes it particularly well suited for the rapid handling of dynamic disrupting events.

Let focus on a significant example: the travel from Duomo to Cascina Gobba.

Moovit provided three different routes:

1. subway M1 (from Duomo to Loreto) + subway M2 (from Loreto to Cascina Gobba) (28mins)
2. subway M1 (from Duomo to Gorla) + bus 44 (from Gorla to Cascina Gobba) (30mins)
3. subway M1 (from Duomo to Precotto) + bus 86 (Precotto to Cascina Gobba) (41mins)

Note that all three solutions share almost half of the path (subway M1 from Duomo to Loreto): any disruption on this segment cannot be bypassed.

Google Maps provides two distinct solutions:

1. Subway M1 (from Duomo to Loreto) + Subway M2 (from Loreto to Cascina Gobba)
2. Subway M3 (from Duomo to Stazione Centrale) + Subway M2 (from Stazione Centrale to Cascina Gobba)

In this case, the shared portion of the path is even longer, accounting for more than a half of the whole trip (subway M2 from Loreto to Cascina Gobba). Note how even the union set of all the solutions provided by Moovit and Google Transit presents the station of Loreto as a common point of failure: should a disruption occur here, the user would have no clue on how to bypass it even querying both the systems.

Let see how the MOTUS planner could handle this situation. First of all, we note how, in normal conditions, the base solution is equivalent to one of Google's.

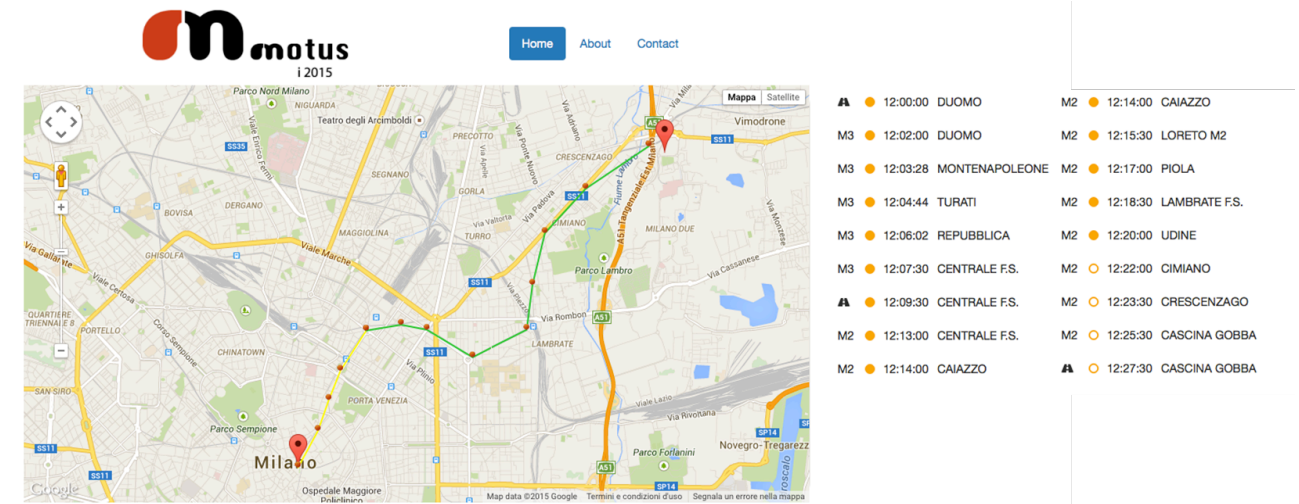


Fig. 6. MOTUS solution from Duomo to Cascina Gobba in normal conditions

If we inject the information that the M2 station of Loreto is unavailable, MOTUS automatically suggests the user to replace the M2 trait with the 56 bus which, even though slightly more slowly, follows the same route.

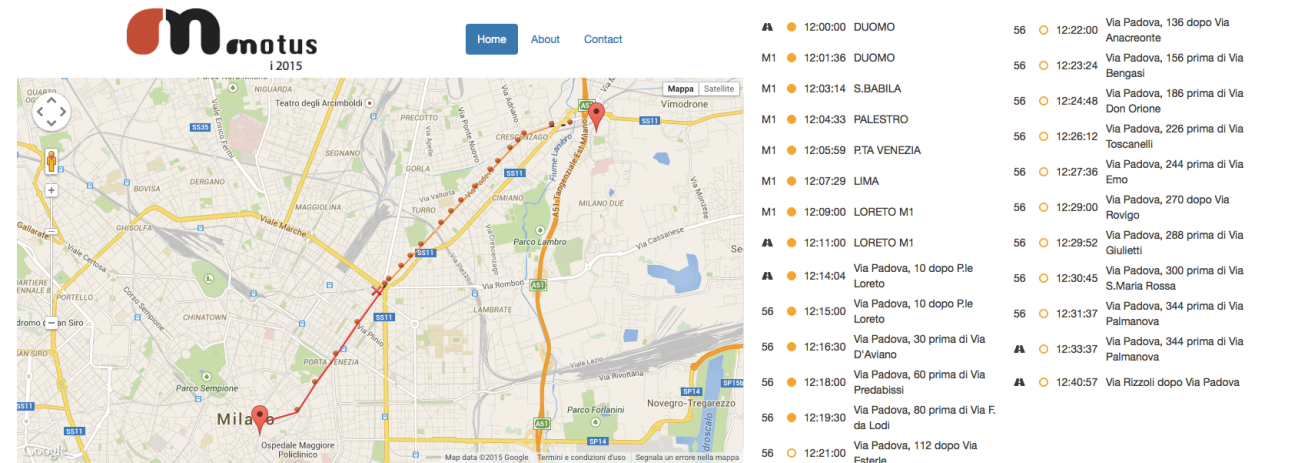


Fig. 7. MOTUS solution from Duomo to Cascina Gobba in case of disruption in Loreto M2

If we mark the Loreto subway stations inapproachable from all lines (both M1 and M2), MOTUS provides a more radical reroute, exploiting line 23 to bypass Loreto and take subway M2 directly from the next station towards Cascina Gobba

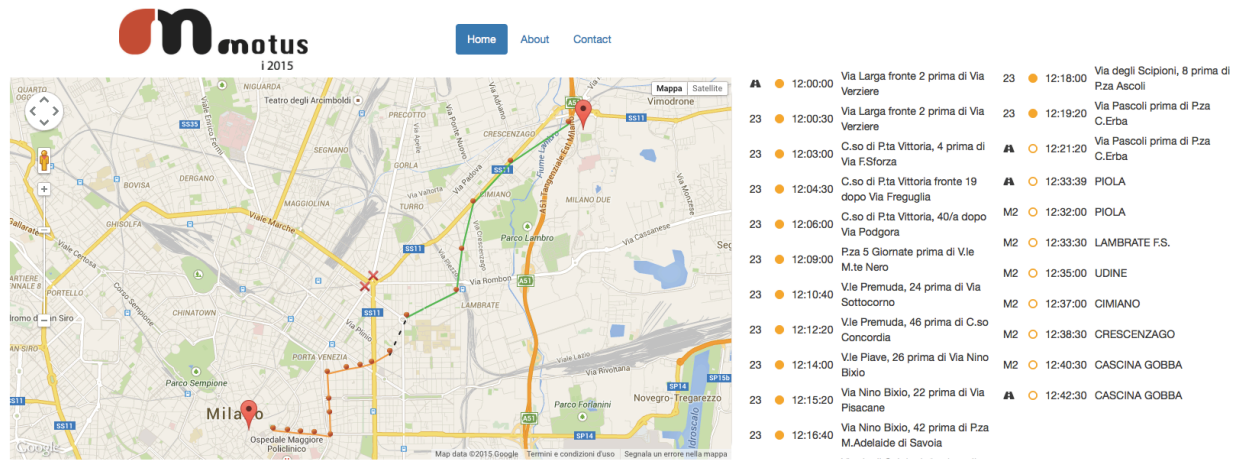


Fig. 8. MOTUS solution from Duomo to Cascina Gobba in case of disruption in Loreto M1 and Loreto M2

As shown in the example, solutions provided by MOTUS algorithm are solid and similar to those provided by other travel planners already available. Furthermore the algorithm allows to dynamically modify the network status, making a flexible management of emergencies/disruptions possible.

#### 4. Conclusions

In this paper we presented a travel planner for the management of unexpected events, delays and service disruptions concerning public transportation. Such a tool allows to model a range of service disruptions, much wider than those possible with approaches currently proposed by transit agencies. For instance a user can ask for a trip not passing through a specific station that is temporarily closed. Some tests on the MOTUS travel planner have been made to compare its performances with those of other famous travel planners like Google Transit and Moovit. Such tests show that, in normal conditions MOTUS yields paths very similar to those proposed by the latter and with the same good CPU time performance of Google Transit (i.e., few tenths of seconds). While, in case of disruption, MOTUS proposes reasonable alternative paths that bypass the disruption and that, most of the times, are not generated by the other travel planners.

One potential impact of the service presented concerns the perception of the transit service by the users: if disruptions are easily and automatically dealt with, the perception of reliability and robustness greatly increments, and the uncertainty reduced, making the transit travel option more attractive.

Future developments of the model will manage multimodal trips, allowing the user – for instance – to organize a trip with LPT and car sharing, and will also provide a set of alternative paths.

#### Acknowledgements

This work was developed within the project MOTUS (MObility and Tourism in Urban Scenarios), co-funded under the programme Industria 2015 (Progetti di Innovazione Industriale - Bando Mobilità Sostenibile) of the Italian Ministry of Economic Development. The authors would like to thank Daniele Bignami of Fondazione Politecnico di Milano for his support to the overall project coordination.



## References

- Arena, M., Azzone, G., Colomi, A., Conte, A., Luè, A., Nocerino, R., 2014. Service design in electric vehicle sharing: evidence from Italy. *IET Intelligent Transport Systems*, 9(2), 145-155.
- Bar-Gera, H., 2007. Evaluation of a cellular phone-based system for measurements of traffic speeds and travel times: A case study from Israel. *Transportation Research Part C: Emerging Technologies*, 15, 6, 380-391.
- Bast, H., Carlsson, E., Eigenwillig, A., Geisberger, R., Harrelson, C., Raychev, V., Viger, F., 2010. Fast Routing in Very Large Public Transportation Networks Using Transfer Patterns. In: *Algorithms – ESA 2010 Lecture Notes in Computer Science Volume 6346*, 2010, pp 290-301.
- Bauer, R., Delling, D., Wagner, D., 2011. Experimental study of speed up techniques for timetable information systems. *Networks*, 57(1), 38–52.
- Böhmová, K., Mihalák, M., Pröger, T., Šrámek, R., Widmayer, P., 2013. Robust Routing in Urban Public Transportation: How to Find Reliable Journeys Based on Past Observations. In *Proceedings of 13th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems (ATMOS'13)*. OpenAccess Series in Informatics, 27–41.
- Bruglieri, M., Colomi, A., Lia, F., Luè, A., 2014. A Multi-objective Time-dependent Route Planner: A Real World Application to Milano City. *Transportation Research Procedia*, 3, 460-469.
- Caccia Dominioni, O., Ciccarelli, D., Colomi, A., Luè, A., 2008. A traffic estimation model based on inductive loops and floating car data for the city of Milano: a work in progress, *Proceedings of the 12th Meeting of the EURO Working Group on Transportation*, Ischia.
- Chorus, C.G., Molin, E.J., van Wee, B., 2006. Travel information as an instrument to change car-drivers' travel choices: A literature review. *European Journal of Transport and Infrastructure Research*, 6, 4, 335-364.
- Ciccarelli, D., Colomi, A., Luè, A., Morrocchi V., Rossi, D., 2006. Integrated information and management system for traditional and demand-responsive mobility services in city of Milan, Italy, *Proceeding of the Transportation Research Board 85th Annual Meeting*, Washington DC.
- Delling, D., Pajor, T., Werneck, R.F., 2012. Round-based public transit routing. *Algorithm Engineering and Experiments (ALENEX)*, 130–140.
- Delling, D., Dibbelt, J., Pajor, T., Wagner, D., Werneck, R. F., 2013. Computing Multimodal Journeys in Practice. In: *Experimental Algorithms, Lecture Notes in Computer Science Volume 7933*, pp 260-271.
- Grotenhuis, J.W., Wiegman, B.W., Rietveld, P., 2007. The desired quality of integrated multimodal travel information in public transport: Customer needs for time and effort savings. *Transport Policy* 14, 1, 27-38.
- Jariyasunant, J., Work, D.B., Kerkez, B., Sengupta, R., Glaser, S., Bayen, A., 2010. Mobile transit trip planning with real-time data,” in *Proceedings of the 2010 Transportation Research Board Annual Meeting*, Washington DC.
- Kolosz, B., Grant-Muller, S., 2015. Extending cost–benefit analysis for the sustainability impact of inter-urban Intelligent Transport Systems. *Environmental Impact Assessment Review* 50, 167-177.
- Luè, A., Colomi, A., Nocerino, R., Paruscio, V., 2012. Green Move: An Innovative Electric Vehicle-Sharing System. *Procedia-Social and Behavioral Sciences*, 48, 2978-2987.
- Luè, A., Colomi, A., 2015. Conflict analysis for environmental impact assessment: a case study of a transportation system in a tourist area. *Group Decision and Negotiation* 24(4), 613-632.
- Maccubbin, R. P., Staples, B., Kabir, F., Lowrance, C.F., Mercer, M.R., Philips, B.H., Gordon, S.R., 2008. Intelligent transportation systems benefits and costs: 2008 update. U.S. Department of Transportation Research and Innovative Technology Administration. Report available at <http://ntl.bts.gov/lib/30000/30400/30466/14412.pdf>
- Müller-Hannemann, M., Schulz, F., Wagner, D., Zaroliagis, C. 2007. Timetable information: Models and algorithms. In *Algorithmic Methods for Railway Optimization*. Springer, 67–90.
- Naniopoulos, A., Bekiaris, B., Panou, M., 2004. Costs and benefits of information technology systems and their application in the infomobility services: The travel-guide approach. *Research in Transportation Economics* 8, 463-480.
- Pallottino S., Scutellà. M. G., 1998. Shortest path algorithms in transportation models: Classical and innovative aspects. In *Equilibrium and Advanced Transportation Modelling*. Kluwer Academic Publishers Group, pp.245–281.
- Pyrga, E., Schulz, F., Wagner, D., and Zaroliagis, C., 2008. Efficient models for timetable information in public transportation systems. *Journal of Experimental Algorithmics*, 12, 2-4.
- Pyrga, E., Schulz, F., Wagner, D., Zaroliagis, C., 2004a. Towards Realistic Modeling of Time-Table Information through the Time-Dependent Approach. *Electronic Notes in Theoretical Computer Science*, 92(2), 85-103.
- Pyrga, E., Schulz, F., Wagner, D., Zaroliagis, C., 2004b. Experimental Comparison of Shortest Path Approaches for Timetable Information. In *Proceedings of the Sixth Workshop on Algorithm Engineering and Experiments (ALENEX)*, Society for Industrial & Applied Mathematics.
- Schulz, F., 2005. Timetable Information and Shortest Paths. Doctoral dissertation, Karlsruhe University. Available at <http://digbib.ubka.uni-karlsruhe.de/volltexte/1000002042>.