shortest path, multiobjective, intermodal networks

Mounir BOUSSEDJRA<sup>1</sup> Christelle BLOCH<sup>1</sup> Abdellah EL MOUDNI<sup>1</sup>

## A MULTIOBJECTIVE INTERMODAL SHORTEST PATH (MOISP)

The paper begins by an introduction, the description of the treated problematic and some definitions of the different concepts (e.g. intermodality, transshipments, travel and waiting times...) used in the following (section 2). A brief survey presents some related works (section 3). The proposed contribution is explained in three main points: the model used to formulate the problem (section 4), the description of a given solution (section 5), and the basic principles of the solving algorithm (section 6). The correctness and the complexity of this contribution are discussed (section 7) and its performances are presented (section 8). Finally, the paper is concluded by some prospects (section 9).

# WIELOZADANIOWA NAJKRÓTSZA ŚCIEŻKA INTERMODALNA

W artykule badany jest problem odnajdowania najkrótszej ścieżki na trasie punkt Wyjścia – punkt Przeznaczenia (O-D) w intermodalnych sieciach transportowych, mających na celu minimalizację czasu podróży i liczby pośrednich przesyłek dla wymaganej ścieżki. Sieć transportowa oraz odnośne dane są modelowane za pomocą wieloetykietowych grafów. Problem intermodalnych najkrótszych ścieżek i jego definicja zostały omówione pokrótce. Zaprezentowano algorytm opracowany do znajdowania ścieżki. Istotnym jest zwłaszcza zwrócenie uwagi na wdrożenie tego podejścia oraz wyników, jakich ono dostarcza.

## 1. INTRODUCTION

Ground transportation is a field in which many problems that can be modelled and solved by mathematical optimization techniques occur. However, the existing works in this field often deal with simplified models, which are far from the complexity of the practice [9]. In particular, shortest path problems (SP) are among the most studied network flow optimization problems [19]. They all aim at finding the shortest path between some points, called "sources" and "destinations", in a given network but they differ according to the

<sup>&</sup>lt;sup>1</sup> Laboratoire Systèmes et Transports (S.e.T), Université de Technologie de Belfort-Montbéliard, 90010 Belfort, FRANCE {mounir.boussedjra,christelle.bloch,abdellah.el-moudni}@utbm.fr

network characteristics (e.g. with or without cycle) and the criterion to be optimized (e.g. the time and/or the cost of a travel). The variation of these parameters is linked to the nature of the tackled applications. For instance, if the studied application involves several sources and several destinations in a static network (i.e. with fixed travel times), the problem is known as an all-to-all static problem. Whereas an instance with a single source and a single destination where travel times are functions of time belongs to the class of one-to-one dynamic problems. In many cases, solving a problem requires the development of adequate solving procedures, in order to take into account its specific constraints while keeping reasonable computation time and memory space. Consequently, this kind of problem has been the subject of many research works, according to the various considered assumptions and constraints [2]. The most recent works take into account a new concept, called intermodality, that brings new interesting constraints to the SP problems. Over and above the dynamic comportment of the network, other parameters are considered, such as waiting times and transshipments between modes. Consequently, the last generation of works tends to solve this type of problem and to propose a multiobjective solution.

### 2. PROBLEMATIC AND DEFINITIONS

An intermodal transportation can be defined as the serial use of different transport modes to move passengers and/or freight from a place to another [11]. In a wider sense, the term intermodal is used to describe a transport system in which at least two modes are required to make a travel between two points. This characteristic, intermodality, involves some additional constraints in the shortest path problem : travel times and/or travel costs do not only depend on source and destination points (in particular on the distance between them) but they also depend on the chosen departure time and the used transport mode. This relation between travel time and departure time makes the transportation network be a dynamic network. With several transport modes, each transfer time (also called switching time) from a mode to another must be considered. In the remainder of the paper, we use a variable called waiting time to represent this parameter. These constraints make the SP problem be more complex, so it belongs to the class of dynamic shortest path problem, [1, 18 and 19], or the class of time-dependent intermodal optimum path problems [18, 20]. The solution complexity varies according to the chosen model and the considered constraints. In this paper we consider three transport modes: bus, car and train. Some modes work on various transport way connections (or lines), and then we must consider each of these lines as a complete mode. The studied network is not a first-in-first-out (FIFO) network, as the travel time on each arc depends on the departure time and on the transport mode used. The departure times, the transport modes available on each node and the corresponding travel times are fixed. The starting time and the finishing time of the travel are also specified before the path research. They represent respectively the minimum departure time from the source point  $Td_s$  and the maximum arrival time to the destination point  $T_{a_D}$ . The maximum number of transshipments for the wished path is also specified before the research. The path is accepted if and only if its travel time is less than or equals the difference between  $Td_S$  and  $Ta_D$  and if its number of transshipments is less than the specified number. The proposed approach aims at minimizing the travel time and the number of transshipments for the required paths between s and D. The motivation consists in the multi-objective nature of the problem and the lack of a general solution for this problem in the intermodal network case. Indeed, this kind of problem is

known to be a NP-hard problem [19]. So, the developed algorithms generally treat simple network models, as it will be shown in the next section, that gives an overview of the literature dedicated to the SPs problems.

#### 3. RELATED WORKS

Most of the main solving algorithms developed to solve shortest path problems [3, 10, 12 and 17] can rarely be directly applied to complicated cases. They deal with the simplest cases of the problem, in which there is a single transport mode and travel times are static. Other recent works focus on the dynamic SP on time-dependent graphs [2, 7, 19 and 20]. Since travel times on this type of networks depend on the departure time in each point, many networks exhibit this kind of dynamic behaviour [8, 19 and 20]. Finally, over the past few years, there has been an increasing interest in the combination of different transportation means to make intermodal transportation systems be more efficient [4, 5, 16 and 20]. Thus, a new class of solution is developed to treat intermodal network problems with specific constraints. The paths are generally characterized by two or more attributes. So, some researches focus on multiobjective shortest path problems (SPs) [4, 14, 15 and 19], with goals of relevant interest, like the minimization of cost, time and number of transshipments. Pallotino and Scutellà indicate that such problems are NP-Hard problems [19]. These authors particularly focus on the bicriterion problem (travel time and number of transshipments). They suppose that, when the list of possible values for one of the two criteria is finite or a priori known, the criterion can be bounded by a maximal value. And so, the solution approach consists in minimizing the second criterion while the first criterion is used as an attribute. In this paper, we propose to solve in the same way the one-to-one intermodal shortest path problem that aims at minimizing the same criteria. The solving approach that we propose differs from [19] in two main points. Firstly, it uses several constraints linked to the problem specification (such as the departure time and the arrival time) to reduce the search space. Secondly, it uses a bidirectional strategy to treat the graph (Section 6).

#### 4. INTERMODAL TRANSPORTATION NETWORK MODEL

The studied transportation network is represented by a graph G = (N, V, T, X) where N is the set of nodes,  $N = \{1...n\}$  representing the stations. V is the set of arcs,  $V = \{(i,j) \in N \times N\}$ , and each arc represents the route segment between each pair of stations (nodes).  $X = \bigcup X(i)$  is the set of the transport modes that serve the network stations and finally  $T = \bigcup T^{X(i)}$  is the set of departure times for all transport modes. Let  $\Gamma_i^-$  (respectively  $\Gamma_i^+$ ) denote the predecessor set (resp. successor set) of *i*. Practically, in each node  $i \in N$  the set of data X(i) is defined. It represents the transport modes available on the node *i* (i.e. that can be used to link it with its successor points  $v \in \Gamma_i^+$ ). This set is represented by  $X(i) = \bigcup X^v(i)$  where for each  $v \in \Gamma_i^+$  we have

 $X^{\nu}(i) = \{x_1^{\nu}(i), \dots, x_k^{\nu}(i)\}$ . Each transport mode  $x_j^{\nu}(i)$  has a set of departure times  $\tau[x_j^{\nu}(i)] = i_1[x_j^{\nu}(i)] \dots i_m[x_j^{\nu}(i)]$ . The latter defines the possible departure times to travel from node *i* to node  $\nu \in \Gamma_i^+$  with the mode  $x_j^{\nu}(i)$ .  $T^{\nu}[X(i)] = \bigcup_j T[x_j^{\nu}(i)]$  is the set of possible departure times on

the node *i* to go towards the node *v* using any of the transport modes available in node *i*. Finally, the parameter  $T^{x(i)} = \bigcup T^{v}[x(i)]$  defines the set of all departure dates possible at node *i* to

go to its successors. The label Tr(i, j) associated with node i equals 1 if a transshipment is needed to go from *i* to *j*,  $j \in \Gamma_i^+$ , it equals 0 otherwise. The difference between the departure time and the arrival time on a given node of the graph defines the waiting time on this node. This value, the waiting time, is also represented by the needed time to perform transshipment between two modes. In destination node D this value equals zero as well as if the departure and the arrival modes on the node are the same. However, on origin s it is equal to the difference between the time i at which the user really leaves s and the wished departure time  $Td_s$ . In the defined model this value represents another label associated with each node. The notation  $\Psi_{lii}^{xy}(t)$  where  $x \in X^{j}(l)$ ,  $y \in X^{i}(j)$  and  $t \in T[y]$  is used to represent it. More precisely,  $\Psi_{lii}^{xy}(t)$  represents the needed time to change from mode x to mode y, on node j when one travels from node l to node i, via j and leaving j at time l. A value of the travel time  $\phi_{ii}(x,t)$  is assigned to each couple (x,t) (where  $x \in X^{j}(t)$  and  $t \in T[x]$ ). It is represented by a label of arc (i, j) and it defines the time needed to travel from node i to node j with the transport mode x leaving i at time i. Thus, each arc carries as many labels as the number of possible modes and starting dates. To illustrate this, the figure 1 represents an intermodal transportation network where the different available modes  $x = x_1^k(i) = x_2^j(i)$ ,  $y = x_1^h(k)$  and  $z = x_1^j(i) = x_2^k(i)$  are represented by different arcs.



Fig.1. Graphic representation of the transportation network

#### 5. DESCRIPTION OF A GIVEN SOLUTION

A solution of the problem, i.e. a given path connecting the origin node s to the destination node D by leaving from s at the date  $t_1$  consists of a sequence of nodes i for which the couples (x,t) of departure, with  $x \in X^{\nu}(t)$  and  $t \in T[x]$ , are fixed.

Thus, the values of  $\phi_{ij}(x,t)$  and  $\Psi_{iji}^{xy}(t)$  can directly be deduced from this path description. The following formulation illustrates this:

$$Path = \{(n_1 = S, x_1, t_1), (n_2, x_2, t_2), ..., (n_j, x_j, t_j), (n_f = D, x_f, t_f))\}$$
  
where  $n_i \in N, x_i \in X^{n_{i+1}}(n_i), t \in T[x_i] \text{ and } n_{i+1} \in r_{n_i}^+$  (1)

The length  $\Pi_{SD}(t_1)$  of this path, i.e. the travel time, is computed by adding the value of  $\phi_{n_in_i+1}(x_i,t_i)$  and  $\Psi_{n_{i-1}n_i,n_{i+1}}^{x_{i-1}x_{i}}(t_i)$  for all of its nodes. The formulation (1) which represents the path allows to specify from the node  $n_i$  its successor  $n_{i+1}$  and its predecessor  $n_{i-1}$  and the corresponding modes and dates. Thus, the travel time value  $\phi_{n_in_i+1}(x_i,t_i)$ , the waiting time value  $\Psi_{n_{i-1}n_i,n_{i+1}}^{x_{i-1}x_{i}}(t_i)$  and the number of transhipment  $Tr(n_i, n_{i+1})$  can be respectively represented by  $\phi(n_i), \Psi(n_i)$  and  $Tr(n_i)$ . Consequently, the calculation of  $\Pi_{SD}(t_1)$  can be represented by the following formulation:

$$\Pi_{SD}(t_1) = \{ (\phi(n_1 = S) + \Psi(n_1)) + (\phi(n_2) + \Psi(n_2)) + \dots + \phi(n_f = D) \}$$
(2)

Thus, the studied SP problem can be formulated as follows: in a graph G(N,V), determine the solution *path* which minimizes the total travel time (including waiting times)  $\Pi_{SD}(t_1)$  computed from the values  $\phi(n_i)$ ,  $\Psi(n_i)$  and  $Tr(n_i)$  for each node  $n_i$  belonging to *path*. In case of equality, the number of transshipments Tr(s,d) is used as a second level of decision Using the above notations, this results in the following formulation:

Minimize 
$$f(path) = \sum (\Psi(n_i) + \phi(n_i))$$
 (3)

$$Minimize h(path) = \sum Tr(n_i)$$
(4)

where  $n_i \in path / n_i = S...D$ .

The objective function (3) minimizes the total travel time, while the objective function (4) minimizes the number of transshipments.

## 6. THE SHORTEST PATH ALGORITHM

This section describes the principles of the multiobjective intermodal shortest path algorithm. This one allows to find the best intermodal path in a directed graph with or without cycle where the arcs have positive lengths. The research is performed according to the travel time, to the number of transshipments and also according to the different constraints of the problem. During the graph exploration, the path is built node-by-node, and the label values associated with nodes are modified using a correcting method based on the same principles as the algorithm that Dijkstra proposed in 1959 [13]. Practically, the algorithm uses a bidirectional treatment strategy: it begins the search of the path from the source and the destination nodes (that will be called search departure nodes SDN in the following) iteratively. At each iteration, the label correcting method changes the label values to solve the Bellman optimality condition [19]. (i.e. the label value is modified if it is greater than the new computed value). It selects, at each iteration, between all eligible nodes (i.e. the nodes for which the path length value is modified by the label correcting method), the node *i* for which the partial travel length ( $\Pi_{si}(t_i)$  or  $\Pi_{i0}(t_i)$ ) is the best. The selection is performed by examining, for all the eligible nodes, all the possible couples (transportation mode, departure time) that connect these nodes with their successor or predecessor nodes (according to the considered SDN) and by choosing the node that minimizes the travel length and the number of transshipments. Thus, the values of travel and waiting times that we use to compute the path length are deduced. All eligible nodes are kept in a successor list SL or in a predecessor list PL according to the SDN. In the beginning, SL and PL respectively contain the nodes S and D. The first iteration consists in updating all nodes that can be directly reached from S or from which D can be directly reached. Consequently, if the considered SDN is S the waiting time is computed in the departure node (the node that belongs to SL). In the second case, when the SDN is D, the waiting time value is computed in the updated nodes (i.e. predecessors of the current node). It is worthy of note that the label denotes the travel time between the considered SDN and the current node in both cases, but it is related to the optimum path in the first case (for a path from i to S) whereas it is simply an estimated value in the second case (for a path from to D). After these steps, the nodes that are reached respectively from s or D and whose label values  $(\prod_{s_i}(t) \text{ or } \prod_{iD}(t))$  are modified are added to SL and PL respectively. In the next step, these nodes are used to update the label value of the remainder nodes of the graph and to build the path. This is performed by selecting at each iteration (from SL or PL alternatively), the node that has the best path length (i.e. the best label value). The policy used in this paper allows us to treat only the paths for which the length value is improved. That is, if two paths are possible to arrive at any node, then we store only the one that minimizes global travel time and number of transshipments. So, all other possible paths that include the node which was not stored are eliminated. Besides, the algorithm also reduces the combinatory of the problem by checking the following constraints: the travel times of the built (partial or total) paths must be less than the difference between  $Ta_D$  and  $Td_s$ , the arrival dates at nodes  $i \in SL$  must be less than or equal to  $Ta_D$ , the departure time of the couples (x, i)associated with PL must be less than the possible arrival dates, the number of transshipments must be less than the specified maximum number of transhipments. If these constraints eliminate all the possible paths, the algorithm detects an infeasibility case. It may also provides this conclusion when no possible path between s and D exists in the graph (which may occur if the graph is not connected). Otherwise, the algorithm stops when SL

(respectively PL) contains D (respectively S), which means that the optimal solution has been reached.

## 7. COMPLEXITY AND CORRECTNESS

The proposed approach involves two search departure nodes and the number of iteration is strongly related to the considered SDN. In the first case, the search departure node is s, for each node i selected from  $SL_{i}[T][X]$  iterations are needed to compute the waiting and the travel times on the nodes that belong to  $\Gamma_{1}^{+}$ . Thus, in the worst case, all the graph nodes belong to  $\Gamma_{2}^{+}$ . and the algorithm complexity is O(|N||M||T|). Practically, the number of successors is bounded for each node by a fixed value, which reduces the complexity to  $O(|\Gamma_i^+||X||T|)$ . In the second case, when the search departure node is D, computing the waiting time in the nodes that belong to PL (and then deducing the travel time and selecting the next node on the path) requires  $|N|^2 |M|^2 |T|^2$  iterations. Indeed, it is necessary to compute the possible arrivals times on the nodes belonging to PL. This requires the enumeration of all the transportation modes and all the departure times associated with the elements of  $\Gamma_{1}^{-}$ . Thus, in the worst case, when all the graph nodes belong to PL and  $\Gamma_i^-$ , the algorithm complexity is  $O(|N|^2 |M|^2 |T|^2)$  when the considered SDN is D. It must be noted that, in both cases, this estimation of complexity is an upper bound: actually, checking the different constraints of the problem permits to greatly decrease the number of variables really treated (i.e. |x| and |t|), which significantly improves the performances of the proposed algorithm.

## 8. EXPERIMENTAL RESULTS

This section presents the tests that have been run on a 2GHz processor to estimate the performances of the algorithm. The considered networks were randomly generated and they contained 100, 500 and 1000 nodes. The number of successors was limited at three for each node, in order to increase the complexity of the problem: indeed, without this constraint, the probability that a direct path from S to D exists is too important and consequently the example is not appropriate to run efficient tests. The number of transportation modes in each node was strictly related to the number of nodes in the graph. For each mode, on each node, there were 200 departure times. The table 1 presents the environment of execution: it contains the number of nodes (Nb-Nodes), the number of transportation modes in each node (NTM) and the number of stations visited by each transportation mode (NSM). The provided results are compared with those returned by a branch and bound (B&B) method. This method searches the local minimum at each iteration. It uses a depth-first search strategy to cover thegraph. The first computed travel length is used to evaluate the next solutions. At each iteration the algorithm checks that both the computed travel length and the number of transshipments are less than the maximum values specified before the search. The table 2 shows the CPU times (in seconds) that represent the results of the execution. The provided values were obtained by averaging the results of ten tests for each size of graph. Each test was performed in a new randomly generated graph while keeping the same source and destination nodes.

The execution environment

Nb-Nodes	NTM	NSM
100	13	8 - 13
500	25	15-25
1000	50	24 - 33

The execution results

Nodes	100	500	1000
MOISP	1,95	5,37	11,92
B&B	8,45	13,2	28,4

#### 9. CONCLUSION

The primary aim of this work was the coordination between different modes of transport. We have developed an algorithm that allows to compute the best intermodal path between two points source and destination while specifying the wished departure and arrival times and the maximum number of transshipments. This is motivated by several folds. The first one is the multi-objective nature of the shortest path problem. The second one consists in the lack of a general solution for this problem in the intermodal network case, in particular due to the running time and the memory space required to solve real instances of the problem. The proposed approach is based on a bi-directional search strategy to explore the graph. Different kinds of constraints (linked to the total travel time and the total number of transshipments) permit to reduce the search space. This allows to reach good performances in comparison with a branch and bound method. Although the results are satisfying, these works can still be improved by studying over multiobjective strategies, such as the search of the pareto optima.

#### BIBLIOGRAPHY

- AHUJA R.K., ORLIN J.B., PALLOTTINO S., SCUTELLA M.G., Minimum time and cost path problems in street networks with traffic lights, rapport technique, TR-99-13.
- [2] AHUJA R.K., ORLIN J.B., PALLOTTINO S., SCUTELLA M.G., Dynamic shortest paths minimizing travel times and costs, in Proceedings of the Mediterranean Electrotechnical Conference, vol. 2, pp. 656-659 CODEN: PMECFA, 2001.
- [3] BELLMAN R.E., On a routing problem. In Quart. Appl. Math, 16: 87-90, 1958.
- [4] BISTARELLI S., MONTANARI U., ROSSI F., in Proc. CP-AI-OR'99 Workshop on Integration of AI and OR techniques in Constraint Programming for Combinatorial Optimization Problems.
- [5] BOILE M. P. Intermodal Transportation Network Analysis A GIS Application. Proceedings of the 10th Mediterranean Electrotechnical Conference, Information Technology and Electrotechnology for the Mediterranean Countries, 11: 660–663, 2000.

Table 2

Table L

- [6] BOUSSEDJRA M., BLOCH C., EL-MOUDNI A., Solution optimale pour la recherche du meilleure chemin intermodal. Proc: 4e Confrence Francophone de MOd'elisation et SIMulation MOSIM, : pp. 230-238, 2003.
- [7] CARLOS F.D., Reversibility of the time-dependent shortest path problem. Transportation Research Part B, 36: 665-668, 2002.
- [8] CHABINI I., Discrete dynamic shortest path problems in transportation application: complexity and algorithms with optimal run time, Transportation Research, pp. 170-175, 1997.
- [9] CORDEAU J.F., TOTH P., VIGO D., A survey of optimization models for train routing and scheduling, Transportation Science Vol. 32, pp. 380-404, 1998.
- [10] DUKSTRA E.W., A note on two problems in connexion with graphs. Numerische Mathematik, 1 : 269–271, 1959.
- [11] FEBBRARO A. D., SACONE S., A new algorithm to finding the best path in intermodal transportation systems, Studies in Informatics and Control, Vol. 4, pp. 357-363. 1995.
- [12] FLOYD R.W., Algorithm 97, Shortest path. Comm. ACM, 5: 345, 1962.
- [13] GONDRAN M., MINOUX M., Graphes et Algorithmes, Editions EYROLLES 1985, Paris.
- [14] LOZANO A., STORCHI G., Shortest viable hyperpath in multimodal networks. Transportation Research Part B 36: 853-874, 2002.
- [15] MODESTI P., SCIOMACHEN A., A utility measure for .nding multiobjective shortest paths in urban multimodal transportation networks. European Journal of Operational Research, 111: 495-508,1998.
- [16] MOHAPATRA P.K.J., DUTTA R.K., An intermodal investment decision model in the transport sector. Omega, 18(2): 203-212, 1990.
- [17] MOORE E.F., The shortest path through a maze. Proc. Int. Symp. On theory of switching, II, April 2-5: 285–292, 1959.
- [18] NACHTIGALL K. Theory and methodologys Time depending shortest-path problems with applications to railway networks. European Journal of Operational Research, 83: 154–166, 1995.
- [19] PALLOTTINO S., SCUTELLA M. G., Shortest Path Algorithms in Transportation Models: Classical and Innovative Aspects. In Equilibrium and Advanced Transportation Modelling, Kluwer, pp. 1245-281, 1998.
- [20] ZILIASKOPOULOS A., WARDELL W., An intermodal optimum path algorithm for multimodal networks with dynamic arc travel times and switching delays, Elsevier Science, European Journal Of Operational Research, 125, pp. 486-502, 2000.

Reviewer: Ph. D. Grzegorz Dydkowski