Università di Pisa

Dipartimento di Informatica

Technical Report : TR-99-13

Minimum time and minimum cost path problems in street networks with traffic lights

R.K. Ahuja, J.B. Orlin, S. Pallottino, M.G. Scutella'

July 13, 1999

ADDR: Corso Italia 40,56125 Pisa,Italy. TEL: +39-50-887111. FAX: +39-50-887226

Minimum time and minimum cost path problems in street networks with traffic lights

R. K. Ahuja, Dept. of Industrial & Systems Engng, University of Florida
J. B. Orlin, Sloan School of Management, MIT, Cambridge MA
S. Pallottino, M.G. Scutellà, Dipartimento di Informatica, Università di Pisa

Abstract

We shall investigate minimum time and minimum cost path problems in street networks regulated by traffic lights. We shall show that the minimum time path problem is polynomially solvable. On the other hand, in general minimum cost path problems are NP-hard. Particular but realistic cases which are polynomially solvable will be discussed.

Introduction

Dynamic Shortest Path Problems often arise in transportation applications. Given a directed graph $G = (N, A)$, in dynamic problems a positive travel time $d_{ij}(t)$ is associated with each arc (i,j) of G with the following meaning: if t is the leaving time from node i, then $t + d_{ij}(t)$ is the arrival time at node j. In addition to the travel time, a time-dependent travel cost $c_{ij}(t)$ can be associated with (i,j), which is the cost of traveling from i to j through (i,j) starting at time t. Furthermore, there is the possibility of waiting at the nodes: a (unit time) waiting cost $w_i(t)$ can be associated with each node i, which gives the (unit time) cost of waiting at i at time t.

Several models have been defined and analyzed in the literature depending on the properties of the travel time and cost functions (e.g. continuous or discrete), on the possibility of waiting at the nodes (e.g. no waiting, waiting at each node, waiting only at the origin nodes), on the presence of time windows associated with the nodes and so on (for a survey about dynamic shortest path problems, focusing on transportation applications, see for instance [Pallottino and Scutellà, 1998]).

In this paper, the case of street networks regulated by traffic lights will be investigated. Street networks can be represented through a directed graph $G = (N, A)$, where $i \in N$ represents a street intersection, and $(i,j) \in A$ represents the street connecting intersection i to intersection j (two-way streets can be represented by arcs connecting the intersections in opposite direction). The situation will be considered in which, at some (possibly all the) intersections, traffic lights regulate the car flow: periodically, the flow from a street entering the intersection toward a street leaving it is interrupted ("red" period) and allowed ("green" period), according to the light timing.

Car movement can be sensibly affected by the traffic lights. In fact, although the lights are installed with the aim of regulating the cross sections, speeding car movements and also preventing congestion, their presence may affect the travel time, and it is generally perceived by the drivers as the cause of an extra cost (in terms of disutility) when they are obliged to stop during the red period.

In the cities with moderate congestion, the *travel time* along an arc can be assumed to be fixed and known "a priori". However, the travel time along any path p_{od} is affected by the waiting times due to the encountered red periods. These times depend on the starting time at the origin o and, consequently, on the arrival times at the intersections regulated by the traffic lights. So, if one looks for the minimum time path from o to d, given a departure time from o, it is necessary to take into account the waiting time spent at the intersections reached during the red periods.

An analogous problem occurs if we associate a traversing *cost* with each arc. In the problems under consideration, this cost represents a driver's disutility measure of traversing the arc, which is not fully related to the travel time. It may depend on various factors such as travel time, distance, fuel consumption, accident risk, stress, and so on. In addition, stopping at a red light can generate a further cost, i.e., a *waiting cost* in the dynamic networks terminology, which can or cannot be related to the waiting time: there may exist drivers who do not like to stop (independently of the stop duration), and their waiting cost is thus represented by a fixed amount; on the other hand, there may be drivers whose waiting cost increases with the stop duration. Finding minimum cost paths implies to evaluate also these costs.

In this paper, we shall investigate both minimum time and minimum cost path problems in street networks with traffic lights. To our knowledge, these dynamic shortest path problems have never been addressed in the literature.

In Section 2, after a review of the main results from the literature about dynamic shortest paths, we shall formulate both problems in terms of shortest path problems on a suitable, dynamic, auxiliary network. In Section 3, we shall describe a strongly polynomial algorithm for the minimum time case. Then, in Section 4, we shall present some polynomially solvable special cases of the minimum cost path problem, which is NP-hard in general. Some remarks will conclude the paper.

2. The problem formulation

2.1. Dynamic shortest paths

Given a directed graph $G = (N, A)$, with $|N| = n$ and $|A| = m$, in dynamic problems a positive *travel time* $d_{ij}(t)$ is associated with each arc (i,j) with the following meaning: if t is a feasible leaving time from node i along the arc (i,j), then $t + d_{ij}(t)$ is the arrival time at node j. In addition to the travel time, a time-dependent *travel cost* $c_{ij}(t)$ can be associated with (i,j), which is the cost of travelling from i to j through (i,j) starting at time t. There is the possibility of waiting at the nodes; in particular, a (unit time) *waiting cost* $w_i(t)$ can be associated with each node i, which gives the (unit time) cost of waiting at i at time t.

Several models have been defined and analyzed in the literature (see for example Orda and Rom, 1990; Orda and Rom, 1991; Pallottino and Scutellà, 1998). Here we shall focus on *discrete models*, i.e., we shall assume that the time variable t varies in a discrete set $T = \{t_1, t_2, \ldots, t_q\}$. This is not a loss of generality in the context under consideration, since, in the transportation field, time "discretisation" is generally performed. We shall also assume that, if t is a *feasible leaving time* from node i along a certain arc (i,j), then $t + d_{ij}(t)$ belongs to T. Only feasible leaving times will be considered in this work; for the sake of the notational simplicity, they will be often referred to simply as leaving times.

Discrete dynamic shortest paths have been efficiently studied and solved by introducing the so-called *Space-Time network* $R = (V, E)$, where $V = \{i_h: i \in N, 1 \le h \le q\}$, and $E = \{(i_h,j_k): (i,j) \in A, t_h + d_{ij}(t_h) = t_k, 1 \le h < k \le q\}$. Each arc (i_h,j_k) is given the cost $c_{ij}(t_h)$. Waiting arcs can be introduced to model waiting at the nodes.

By construction, R is a standard graph, with $|V| = nq$ and $|E| \le (m+n)q$. Its size is thus pseudopolynomial with respect to the size of the original graph G. Moreover, R has no cycles by construction. In particular, each chronological-type visit of the nodes of R, i.e., a visit for non-decreasing values of the time index which is associated with the nodes of R, provides a topological visit of R.

By exploiting such a property, i.e., by performing a chronological-type visit of the Space-Time network, it is possible to solve dynamic shortest path problems working on R. In particular, the *minimum cost dynamic path problem*, which looks for a minimum cost path from a given origin o to each other node i, given a leaving time, say t_1, from o, can be solved by selecting the nodes of (and thus generating) R by means of a chronological order, i.e., by considering first the nodes corresponding to time t_1, then the ones for t_2 and so on, in $O(|E|) = O(mq)$ time. The resulting algorithmic paradigm, called *Chrono-SPT*, uses a bucket-list $B = \{B_1, B_2, \ldots, B_q\}$ in order to efficiently perform the selection operations, i.e., for implementing the chronological visit of R, where B_h denotes the bucket containing the nodes to be visited at the time instant t_h, $h = 1, \ldots, q$. In such a paradigm, a cost label is maintained for each node i_h of R, which represents the current minimum cost from node o_1 to node i_h. When all the buckets are empty, the minimum of the labels associated with the nodes of type i_h gives the minimum path cost from o to i in G. Observe that, in general, it is not necessary to generate the entire network R a priori, and then apply the algorithmic paradigm, but only the portion of R needed for the minimum cost path computation needs to be generated (*non-redundant portion* of R). Moreover, from a practical point of view, the algorithm can work directly on the input graph G, by treating R implicitly.

An important special case is the *minimum time dynamic path problem*, in which we want to find the paths from a given origin o to any other node i having the minimum arrival time at i, given a departure time, say t_1, from o (a related problem looks for minimum arrival time paths for all the possible departure times from o). In this case, if we look at the Space-Time network, the problem can be reformulated as a "connectivity" problem on R. In fact, for any destination i, we want to find the minimum time instant $t_h \in T$ for which a directed path exists in R from node o_1 to node i_h. This problem, too, can be efficiently solved by a modification of the paradigm *Chrono-SPT*, in which the stop condition is verified as soon as one node of type i_h, for each node i of G, is selected from the bucket data structure.

The minimum time dynamic path problem has generally the same time complexity as the one relative to minimum cost dynamic path computations. However, if the following local property is satisfied, then a stronger approach can be derived.

Let an arc $(i,j) \in A$ be called a *FIFO arc* if leaving i earlier guarantees to arrive no later at j along (i,j) (see, for instance, [Ziliaskopoulos, 1994]). More formally:

$$t_h + d_{ij}(t_h) \leq t_k + d_{ij}(t_k), \text{ for any } t_h < t_k, \qquad (\textit{FIFO property}).$$

Then G is said to be a *FIFO graph* when all its arcs are FIFO, and a *non-FIFO graph* otherwise.

If G is a FIFO graph, then it is easy to show that the minimum time dynamic path problem can be solved by associating only one label with each node i of G, representing the minimum arrival time at i (see, for example, Pallottino and Scutellà, 1998). In this case, any classical shortest path approach can be applied directly to G (in particular, *Chrono-SPT* reduces to the classical Dial's shortest path algorithm with bucket-list [Dial, 1969]).

Related dynamic path problems are the *shortest path problems with time windows*. In these problems, time windows can be associated with the nodes when arrivals and/or departures are allowed only within given time intervals. Let $[a_i, b_i]$ be the *time window* associated with node i, where both a_i and b_i belong to the given time horizon T. Two situations have been addressed in the literature: arrivals and departures from i are allowed only in time instants belonging to $[a_i, b_i]$ (*hard time windows*), either waiting is allowed if

the arrival at i is before time a_i, whereas departures are allowed only in time instants belonging to $[a_i, b_i]$ (*soft time windows*). For instance, in the problem described in [Desrosiers, Pelletier and Soumis, 1983], a constant cost c_{ij} and a constant positive travel time d_{ij} are associated with each arc (i,j), and waiting is allowed if the arrival at i is before time a_i, for each node i. Such dynamic path problems, too, can be easily described in terms of suitably defined Space-Time networks. An adaptation of *Chrono-SPT* has been proposed in [Desrochers and Soumis, 1988].

2.2. The case of street networks with traffic lights

Let the input graph G = (N, A) represent a street network, i.e., i __N represents a street intersection, and (i,j) __A represents the street connecting intersection i to intersection j. Since in transportation networks the number of streets entering or leaving an intersection is usually limited (3 or 4), we can consider G as a sparse graph, i.e., m is O(n).

As in the dynamic networks reviewed in Section 2.1, assume that two measures are associated with each arc (i,j) __A, namely a (constant) travel time dij (>0) and a (constant) travel cost cij (≥0). Furthermore, assume that at some of the nodes, possibly all, traffic lights regulate the traffic by allowing or delaying the vehicles movement according to a pre-defined timing.

More specifically, let i __N be a node regulated by a traffic light, and a = (h,i) and b = (i,j) be two arcs entering and leaving i, respectively, such that it is possible to enter b coming from a. Then, the pair (a,b(is said to be feasible. The traffic regulation or timing relative to (a,b(is a repeated sequence of green and red phases defined on a given, discrete, time horizon T = {t1, …, tq}, and it is completely defined by the triplet [g(a,b), r(a,b), s(a,b)], where:

- $g(a,b)$ is the duration of the green phase;
- $r(a,b)$ is the duration of the red phase;
- $s(a,b)$ is the *phasing value*, i.e., the time distance between the starting of the first green phase and the beginning of the time horizon, t_1: if, at time t_1, $\langle a,b \rangle$ is in green phase, then it is $s(a,b) \leq 0$; otherwise, i.e. it is in the red phase, then $s(a,b) > 0$; in both cases, the first green phase begins at time $t_1 + s(a,b)$.

Note that the timing is defined for pairs of subsequent arcs since, in general, traffic lights may provide different duration of the green and of the red phases for the same entering arc, depending on the leaving arc. Note also that the green phases can be interpreted as periodic time windows (closed at their left end and open at the right end, when strictly positive), over the time horizon T. As in the case of standard dynamic networks with time windows, we shall assume that both ends of all the green phases, falling in the given time horizon, belong to T.

Particular cases can be easily formulated via the triplets. For example, if a node i is not regulated by a traffic light, we can associate the triplet $[t_q - t_1, 0, 0]$ with each feasible pair $\langle a,b \rangle$ relative to i, since each vehicle can pass through i when it arrives at the intersection. Also, an infeasible pair $\langle a,b \rangle$ (i.e., a forbidden movement from arc a towards arc b) can be modeled by the triplet $[0, t_q - t_1, t_q - t_1]$, since it can be viewed as a unique red phase.

An example

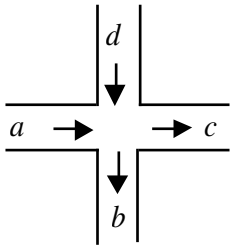

Figure 1 - An intersection

Let us consider two one-way streets crossing at an intersection which is regulated by a traffic light (see Figure 1), and a vehicle moving along the arc *a*.

Let the time horizon *T* be {0, 1,…, *q*}, where times are expressed in seconds. Assume that the timings relative to the arc *a* are defined as follows:

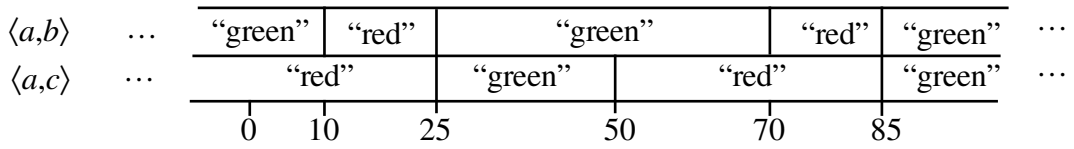

Figure 2 - green and red phases for ⟨*a*,*b*⟩ and ⟨*a*,*c*⟩

Let us define the triplet relative to ⟨*a*,*c*⟩. The first green phase starts at time 25 and it has duration 25, while the duration of the red phase is 35. Then it is [*g*(*a*,*c*), *r*(*a*,*c*), *s*(*a*,*c*)] = [25, 35, 25]. As far as the triplet relative to ⟨*a*,*b*⟩ is concerned, the phasing value *s*(*a*,*b*) is negative since, at time 0, the light is green: since the green duration is 45 and the first green phase ends at time 10, it is *s*(*a*,*b*) = -35. The duration of the red phase is 15, and so [*g*(*a*,*b*), *r*(*a*,*b*), *s*(*a*,*b*)] = [45, 15, -35]. On the other hand, the triplet relative to the infeasible pair ⟨*a*,*d*⟩ is [0, *q*, *q*].

If we want to define path problems on the street network *G*, a crucial point is to compute the waiting time, say $w(a,b,\bar{t})$, of a vehicle arriving at a certain intersection ⟨*a*,*b*⟩ at a time instant $\bar{t}$ (i.e., given the triplet relative to ⟨*a*,*b*⟩, we want to discover whether the phase is either green or red at time $\bar{t}$, and compute the waiting time before leaving the intersection along street *b*). Such a computation can be performed in *O*(1) time by the simple algorithm described below.

Let $\pi(a,b) = g(a,b) + r(a,b)$ be the *period* of the light relative to the pair ⟨*a*,*b*⟩, and define $\theta = \bar{t} \bmod \pi(a,b)$. Two cases have to be considered:

Case 1: Time t_1 is in the red phase, i.e., $s(a,b) > 0$; in such a case it is:

$$w(a,b,\bar{t}) = \begin{cases} s(a,b) - \theta, & \text{if } 0 \le \theta < s(a,b); \\ 0, & \text{if } s(a,b) \le \theta < g(a,b) + s(a,b); \\ \pi(a,b) + s(a,b) - \theta, & \text{if } g(a,b) + s(a,b) \le \theta < \pi(a,b). \end{cases}$$

Case 2. Time t_1 is in the green phase, i.e., $s(a,b) \le 0$; in such a case the following relationships hold true, where the last one is significant only when $s(a,b) < 0$:

$$w(a,b,\bar{t}) = \begin{cases} 0, & \text{if } 0 \le \theta < g(a,b) + s(a,b); \\ \pi(a,b) + s(a,b) - \theta, & \text{if } g(a,b) + s(a,b) \le \theta < \pi(a,b) + s(a,b); \\ 0, & \text{if } \pi(a,b) + s(a,b) \le \theta < \pi(a,b). \end{cases}$$

As an example, consider the intersection in Figure 1, whose timings have been described in Figure 2, and assume $\bar{t} = 1565$. Since $\pi(a,b) = \pi(a,c) = 60$, in both cases it is $\theta = 5$. First consider $w(a,c,\bar{t})$: being $s(a,c) = 25 > 0$, it is $w(a,c,\bar{t}) = s(a,c) - \theta = 20$. Consider now $w(a,b,\bar{t})$: since $s(a,b) = -35 < 0$ and $0 < \theta < g(a,b) + s(a,b) = 10$, it is $w(a,b,\bar{t}) = 0$.

A further concept we need to introduce is the concept of waiting penalty. Given a feasible pair $\langle a,b \rangle$ and an arrival time $\bar{t}$ at the end of street a, let us define the *waiting penalty* function $\gamma(a,b,\bar{t})$ as:

$$\gamma(a,b,\bar{t}) = \begin{cases} 0, & \text{if } w(a,b,\bar{t}) = 0; \\ f(w(a,b,\bar{t})), & \text{if } w(a,b,\bar{t}) > 0; \end{cases} \tag{1}$$

where $f(.)$ is a suitable function representing the disutility for the driver of stopping at a red light.

In this work, *linear waiting penalties* will be addressed, which depend linearly on the waiting time spent at the intersection, plus an additional factor β:

$$f(w(a,b,\bar{t})) = \alpha w(a,b,\bar{t}) + \beta, \tag{2}$$

where α and β are given nonnegative factors.

2.3. The path problem formulation: *the auxiliary network*

In order to formulate and solve path problems from a given origin o to a given destination d in the street graph $G = (N, A)$, it will be often useful to look at an *auxiliary dynamic network* $G^* = (N^*, A^*)$, already introduced by Kirby and Potts (1969) for the case of shortest paths with turn penalties and prohibitions. In our case, such a network is defined as follows:

$$N^* = A \cup \{o, d\}; \tag{3}$$

$$\begin{aligned} A^* = {} & \{(a,b): a, b \in A \text{ and } \langle a,b \rangle \text{ is a feasible pair}\} \\ & \cup \{(o,a): a \in FS(o) \text{ in } G\} \cup \{(a,d): a \in BS(d) \text{ in } G\}, \end{aligned} \tag{4}$$

where $FS(i)$ and $BS(i)$ denote the *forward star* and the *backward star* of $i \in N$, respectively, i.e., $FS(i) = \{(i,j) \in A\}$ and $BS(i) = \{(j,i) \in A\}$.

Intuitively, each node of the auxiliary network, with the exception of o and d, corresponds to an arc of the original graph, and each arc of the auxiliary network represents a pair of feasible arcs of the input graph, plus (o,a) for $a \in FS(o)$ and (a,d) for $a \in BS(d)$. Observe that, by construction, it is $|N^*| = n^* = m+2$; moreover, by our previous assumptions, the number of arcs emanating from each node of G^* is a constant (3 or 4), i.e., $|A^*| = m^*$ is $O(m)$ too. Being $m = O(n)$, both n^* and m^* are $O(n)$, i.e., the size of G^* is only a small constant factor larger than the size of the original graph G.

Let us define now the travel time and the travel cost associated with the arcs of the auxiliary dynamic network G^*.

Given an arc $e = (a,b) \in A^*$ and a leaving time $t \in T$, let the *travel time*, $\tau_e(t)$, be the time needed to go from the beginning of the arc a in G, starting at time t, to the beginning of the arc b in G; so, it is given by the (constant) travel time associated with a in G, i.e., d_a, plus the waiting time at the intersection between a and b, i.e., $w(a,b,t+d_a)$. Note that this travel time depends on the leaving time t.

Clearly, in the case of arcs of type $e = (o,a)$, $a \in FS(o)$, it is $\tau_e(t) = 0$ for each $t \in T$; on the other hand, for any arc of type $e = (a,d)$, with $a \in BS(d)$, it is $\tau_e(t) = d_a$, $t \in T$.

Similarly, as far as the *travel costs* are concerned, given an arc $e = (a,b) \in A^*$ and a leaving time $t \in T$, the travel cost, $c_e(t)$, is given by the (constant) travel cost associated with a in G, i.e., c_a, plus the waiting penalty to be paid at the intersection between a and b, i.e., $\gamma(a,b,t+d_a)$. Clearly, this cost is always zero in the case of the arcs of type (o,a), and it is equal to c_a in the case of arcs of type $e = (a,d)$, $a \in BS(d)$.

Given a leaving time t, it is easy to verify that there is a one-to-one correspondence between the set of paths from o to d in the input street network G, starting from o at time t and composed by sequences of feasible pairs of arcs, and the paths from o to d in the auxiliary dynamic network G^* starting from o at the same time instant t. Moreover, pairs of corresponding paths have the same total travel time (i.e., sum of the travel and waiting times along the path) and the same total travel cost (i.e., sum of the travel costs and waiting penalties along the path).

Therefore, minimum time and minimum cost path problems in G can be equivalently formulated and solved in terms of the auxiliary network G^*.

Specifically, the *minimum time path problem* consists of finding a path from o to d in G (G^*), given a starting time from o, which has the minimum total travel time; the *minimum cost path problem* is to find a minimum cost path from o to d in G (G^*), given the leaving time from o. These problems will be investigated in the following sections where, at least from an algorithmic point of view, it will be often preferable to consider the problem formulations in terms of the auxiliary network G^*; in practice, however, G^* can be managed implicitly by all the proposed algorithmic approaches, in order to achieve improved computational efficiency.

3. The minimum time path problem

Consider the auxiliary dynamic network G^*. Assume to want to find a path from the origin o to the destination d which has the minimum total travel time, provided a leaving time from o.

The following property, which generalizes a similar property observed in [Desrosiers, Pelletier and Soumis, 1983] for the case of single time windows, can be shown to be true.

Lemma 3.1 : G^* is a FIFO graph.

Proof: Consider an arc $e = (a,b) \in A^*$, and a pair of time instants in T, say t and t', such that $t < t'$.

In order to prove the FIFO property, it must be shown that $t + \tau_e(t) \le t' + \tau_e(t')$ or, equivalently, $t + d_a + w(a,b,t+d_a) \le t' + d_a + w(a,b,t'+d_a)$, i.e., $t + w(a,b,t+d_a) \le t' + w(a,b,t'+d_a)$, being d_a a constant.

Observe that, by construction of G^*, $t + d_a$ and $t' + d_a$ represent the arrival times at the intersection $\langle a,b \rangle$ of the input graph G starting from the beginning of street a at time t and at time t', respectively. The following cases have to be considered:

if both $t + d_a$ and $t' + d_a$ belong to a green phase relative to $\langle a,b \rangle$, then both $w(a,b,t+d_a)$ and $w(a,b,t'+d_a)$ are zero, and so the property is verified;

otherwise, let $t' + d_a$ belong to a green phase, whereas $t + d_a$ belong to a red phase: leaving the intersection at time $t' + d_a$ is thus admitted, whereas leaving at time $t + d_a$ is forbidden (i.e., $w(a,b,t'+d_a) = 0$ and $w(a,b,t+d_a) > 0$); anyway, starting at time t, it will be possible to leave the intersection $\langle a,b \rangle$ at a time t^* such that $t + d_a < t^* \le t' + d_a$, and therefore $w(a,b,t+d_a) \le t' - t$. It follows that the FIFO property is verified also in this case.

Similar proofs hold when $t + d_a$ belongs to a green phase and $t' + d_a$ belongs to a red phase, and when both $t + d_a$ and $t' + d_a$ belong to a red phase. ♠

The following result can thus be established:

Theorem 3.1 : The minimum time path problem is solvable in strongly polynomial time.

Proof: From Lemma 3.1 (and due to the results reviewed in 2.1), the minimum time path problem can be solved by standard shortest path algorithms on G^*. ♠

In particular, by using the Fibonacci heap implementation of classical Dijkstra's approach, the problem can be solved in $O(m^*\log n^*)$ time, i.e., in $O(n\log n)$ time. Classical label-correcting approaches, generally faster than the label-setting ones from a computational point of view, can be used as well, by allowing one to solve the problem in $O(m^*n^*) = O(n^2)$ time (see, for example, Ahuja, Magnanti and Orlin [1993]).

4. The minimum cost path problem

The problem of finding a minimum cost path from o to d in G (or, equivalently, in the auxiliary network G^*), given a starting time, is NP-hard in general. Consider in fact the resource constrained shortest path problem described in [Desrochers, 1988]. Given a graph, given L resources, and given a resource feasibility window $[a^l_i, b^l_i]$ for each node i and each resource l, and a resource consumption d^l_{ij} for each arc (i, j) and each resource l, the resource constrained shortest path problem consists of finding a minimum cost path from a given origin to a given destination such that, for each node i along the path, the total consumption of resource l falls in $[a^l_i, b^l_i]$, l=1, 2, …, L. When interpreting the resource constraints associated with the nodes as multiple (soft) time windows, then it is easy to see that the resource constrained shortest path problem is a special case of the minimum cost path problem under consideration. The resource constrained shortest path problem has been proved to be NP-hard by generalization of the multi-dimensional knapsack problem, which is known to be NP-hard. Therefore:

Theorem 4.1 : The minimum cost path problem is NP-hard.

As a further observation, consider the subset of street networks such that each penalty waiting function $\alpha w(a,b,\bar{t}) + \beta$ is very high with respect to the travel costs, for each feasible pair $\langle a,b\rangle$ (e.g. β is a very large number). Then, it is immediate to verify that the minimum cost path problem on a street network G belonging to such a special subset is equivalent to the problem of finding a minimum cost path from o to d among the paths involving no waiting, i.e., in other words, arrivals and departures are allowed only within the time windows associated with the nodes. In the case of a single time window for each node, the problem thus reduces to a standard minimum cost path problem in the presence of hard time windows, which is known to be NP-hard (Desrochers and Soumis, 1988).

4.1. Some tractable cases

In the following, two different scenarios will be addressed, which allow one to solve the minimum cost path problem in polynomial time.

First scenario. Given an arc a of the input network G, let us impose the following restrictions on the form of travel and waiting costs:

Assumption 1. The travel cost associated with each arc a is of the form:

$$c_a = \alpha d_a + \sigma(a);$$

moreover, the penalty due to waiting $w(a,b,\bar{t})$ units of time at the intersection between a pair of arcs a and b is the special case of (1) and (2) where $\beta = 0$:

$$\gamma(a,b,\bar{t}) = \begin{cases} 0, & \text{if } w(a,b,\bar{t}) = 0; \\ \alpha w(a,b,\bar{t}), & \text{if } w(a,b,\bar{t}) > 0. \end{cases} \tag{5}$$

In the above linear forms, α has to be considered as a "scale coefficient" time-to-cost, denoting how the time spent to traverse an arc (or wait at a node) is perceived by the user in terms of cost to be paid. On the other hand, the nonnegative coefficient $\sigma(a)$ has to be intended as a cost penalty due to factors other than time, and it can be expressed in terms of the same "scale coefficient" α, in view of the following further assumption:

Assumption 2. $\sigma(a) \le \rho\alpha$ for each arc a, where ρ is a given positive constant.

Lemma 4.2 : Given a starting time from o, under Assumptions 1 and 2 there exists an *elementary* minimum cost path from o to any node v in G, i.e., a path without arc repetition.

Proof: By contradiction. Assume that each minimum cost path from o to a certain node v, starting at the given time instant, is not elementary, and consider one of them, say P, containing two occurrences of a certain arc $a = (i,j)$. Node j is thus visited at a certain time instant t', and then again, at a time instant $t'' > t'$. Let b denote the arc traversed after the second occurrence of arc a. Then path P can be decomposed into a path segment arriving at node j at time t', into a cycle, say C, going from j at time t' to j at time t'', and into a path segment, say P_b, starting with the occurrence of b.

By our assumptions, the cost of C is $\ge \alpha(t''-t') + |C|\sigma_{\min}$, where:

$$\sigma_{\min} = \min\{\sigma(a): a \in A\}.$$

Consider now the cost of the alternative path from o to v which contains only the first occurrence of a, i.e., the path which arrives at j at time t', possibly waits at node j, and then leaves from j along P_b at a time instant $t \le t''$ (observe that the latter property holds true since the pair $\langle a,b \rangle$ has a unique triplet associated with it). The cost of waiting at j in such an alternative path is $\le \alpha(t''-t')$, i.e. it is less than or equal to the cost of C.

Since moving along P_b in this alternative path costs less than (or has the same cost as) moving along P_b in P, it follows that the alternative, elementary, path is a minimum cost path from o to v. ♠

Unfortunately, the so-called *concatenation property*, which would allow one the use of standard shortest path approaches for the problem solution, may not hold, i. e., given a minimum cost path P from o to d, there may exist a subpath P' of P, from o to an intermediate node v, which is not a minimum cost path from o to v. An example is provided below.

Example

Consider the instance in Figure 3, where the values $(\sigma(a), d_a)$ are associated with each arc a. Let the leaving time from o be $t_1 = 0$, and assume $\alpha = 1$.

As far as the light timings are concerned, let the triplets associated with the pairs of arcs be: $\langle b,c \rangle$: (5, 5, 0); the others: (10, 5, 0).

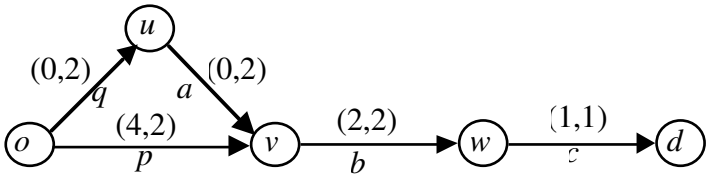

Figure 3 – A counterexample of the *concatenation property*

The minimum cost path from o to d is $P = \{p, b, c\}$, having a cost equal to 12; the alternative path, in fact, arrives at node w at time 6, i.e., during the red period, and it has an over cost of 4 units due to the waiting time. However, the subpath $\{p\}$ of P has a cost equal to 6, which is higher than the cost of the path $\{q, a\}$, which is equal to 4.

In order to overcome such a difficulty, let us introduce a further assumption in the considered scenario, which is indeed realistic:

Assumption 3: For each node and for each street intersection of G, the red phase duration is bounded from above by a given constant ε.

Now, consider the auxiliary dynamic network G^*, and apply the algorithmic paradigm *Chrono-SPT* reviewed in Section 2.1, which allows one to solve general minimum cost path problems in dynamic networks. Then the following property holds true.

Lemma 4.3 : If the Assumptions from 1 to 3 hold true, then the minimum cost path problem can be solved by *Chrono-SPT* by maintaining $m(\rho+\varepsilon)$ different cost labels at most for each node a of the auxiliary dynamic network G^* (i.e., at most $m(\rho+\varepsilon)$ nodes of type a_h have to be generated in the non-redundant portion of the associated Space-Time network).

Proof: Let us consider a minimum cost path, say P^*, from the origin o to the destination d in G^*, leaving from o at the given starting time, which can be assumed to be 0 for the sake of simplicity.

If we denote by a a generic node internal to P^*, then path P^* can be splitted into two subpaths, say P_1 and P_2, where P_1 arrives at a (i.e., at the beginning of the arc a of G) at a certain time instant t'.

Let C_a be the cost of the minimum time path, say P, from o to a, starting at the same leaving time from o, and denote by t such a minimum arrival time at a.

Being P a minimum time path, it is $t' \geq t$, as illustrated in Figure 4. Observe that, by Lemmas 3.1 and 4.2, both P and P^* can be assumed to be paths without repetition of nodes (i.e., without arc repetition in G).

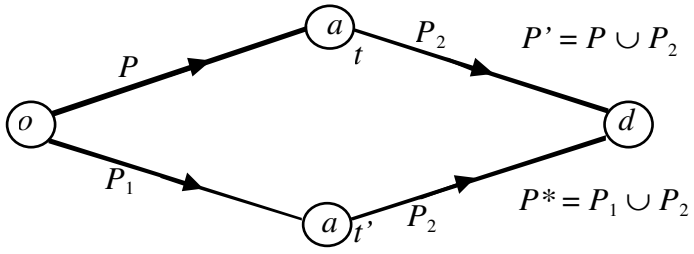

Figure 4 - Comparison between the paths P' and P^*

Now, let us denote by $P' = P \cup P_2$ the path from o to d arriving at node a at time t via the subpath P. Let us compare the two paths P' and P^*, and prove that P^* may have a cost lower than or equal to the cost of P' only if $t' \leq t + m(\rho+\varepsilon)$.

Recall that, given an arc $e = (a,b) \in A^*$, its travel cost, $c_e(t)$, is the sum of the (constant) travel cost associated with a in G, i.e. c_a, and of the waiting penalty to be paid at the intersection between a and b, i.e., $\gamma(a,b,t+d_a)$.

As far as the subpath P_2 is concerned, the sum of the first cost components of its arcs (i.e. the ones of type c_a) is the same in P' and in P^*. On the other hand, the contribution due to the waiting penalties may be different in the two paths; in any case, by Assumption 3, the difference between the time spent in waiting in P_2 by looking at P' and the time spent in waiting in P_2 by looking at P^* is $m\varepsilon$ at most, being P_2 a path without node repetition.

Compare now the path fragments P and P_1. As far as P is concerned, an upper bound to its cost is given by $\alpha t + \alpha m\rho$ (by Assumptions 1 and 2). On the other hand, a lower bound to the cost of P_1 is given by $\alpha t'$.

Now, it is possible to derive necessary conditions under which the cost of P^* is lower than or equal to the cost of P':

$$\alpha t' \leq \alpha t + \alpha m\rho + \alpha m\varepsilon,$$

alternatively,

$$t' \leq t + m(\rho+\varepsilon).$$

Therefore, at most $m(\rho+\varepsilon)$ different cost labels, for at most $m(\rho+\varepsilon)$ different time instants, need to be associated with node a in the Space-Time network in order to compute a minimum cost path (i.e., one for each of the $m(\rho+\varepsilon)$ time instants following the minimum arrival time at a). ♠

By the results reported in Section 2.1 it follows that the non-redundant portion of the Space-Time network associated with G^*, to be generated to solve the minimum cost path problem via a chronological-type approach, has $O(n^*m(\rho+\varepsilon))$, i.e. $O(n^2(\rho+\varepsilon))$, nodes and arcs. Being ρ and ε constant, the problem can thus be solved in $O(n^2)$ time.

Second scenario. In this second case, we do not impose any restrictions on the form of the travel costs and of the waiting penalties, but on the features of the traffic regulation. More specifically, we assume that all the pairs $\langle a,b \rangle$ of G are characterized by the same period over a suitably large time horizon, i.e., the sum of the duration of their green and red phases is the same (these assumptions are quite realistic in the context under consideration):

Assumption 4: $\pi(a,b) = \varphi$ for each pair $\langle a,b \rangle$, where $\varphi > 0$.

Assumption 5: the time horizon T is suitably large.

Then, if we consider the auxiliary dynamic network G^*, the following property holds true:

Lemma 4.4 - Under the Assumptions 4 and 5, the minimum cost path problem can be solved by maintaining at most φ different cost labels for each node a of G^*.

Proof: Consider two paths, say P_1 and P_2, from the origin o to a certain node a of G^* (i.e. a street of G), leaving from o at the time required by the problem formulation. Assume that P_1 arrives at a (i.e., at the beginning of street a) at a time t, $t_1 \leq t < t_1 + \phi$,

whereas P_2 arrives at a at a time $t' = t + \mu\varphi$, where μ is a positive integer (i.e., t and t' differ for a multiple μ of the common period φ). This situation is depicted in Figure 5.

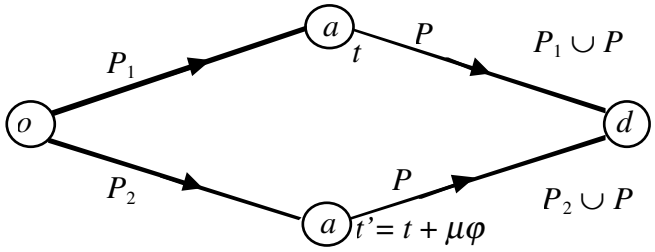

Figure 5 - Comparison between the paths P_1 and P_2

Now, consider a path P of G^* from node a to the destination d, and assume to concatenate both P_1 and P_2 to P, thus obtaining two alternative paths from o to d.

Since, by Assumption 4, all the intersections are characterized by the same period φ, leaving from a along P at time t or at time $t' = t + \mu\varphi$ requires the same total travel time and the same total travel cost to reach the destination d. Moreover, due to Assumption 5, in both cases we are sure to arrive at node d in a time instant belonging to the feasible time horizon T. Therefore, only the minimum between the costs of paths P_1 and P_2 needs to be maintained in order to compute a minimum cost path from o to d having node a as its intermediate node.

More generally, given t, $t_1 \leq t < t_1 + \varphi$, it is sufficient to maintain only the minimum cost label relative to the paths of G^*, from the origin o to the considered node a, arriving at a at times of form $t + \mu\varphi$, where μ is a nonnegative integer.

At most φ different cost labels, for $t_1 \leq t < t_1 + \varphi$, need thus to be associated with each node a of G^*. ♠

The above result provides a bound on the number of cost labels to be maintained for each node a of G^*. However, given t, $t_1 \leq t < t_1 + \varphi$, there is no guarantee that the minimum cost label relative to the paths from o to a, arriving at a at times of the form $t + \mu\varphi$, corresponds to the minimum among such time instants. A chronological-type approach seems thus not suitable for efficiently computing the minimum cost label relative to a, for each a of G^*. Instead, consider a label-setting approach, and apply it to the (non-redundant portion of the) Space-Time network associated with G^*. It is easy to show that, by this kind of approach, for each node a of G^*, once a cost label relative to a time instant of the form $t + \mu\varphi$ is selected by the algorithm, then such a cost label is the desired minimum one. Therefore, only one node needs to be generated in the Space-Time network, for each node a and for each time instant t such that $t_1 \leq t < t_1+\varphi$.

The problem is thus solvable by a standard label-setting approach, like the Fibonacci heap implementation of Dijkstra's algorithm, in $O(n^*\varphi \log(n^*\varphi)) = O(n\varphi \log(n\varphi))$ time.

Then, if φ is a polynomial with respect to the problem size n, the minimum cost path problem is polynomially solvable. In particular, if φ is a constant, as happens in practical applications, the problem time complexity becomes $O(n\log n)$.

Conclusions

This work addresses path problems occurring very frequently in transportation applications, i.e., minimum time and minimum cost path problems in the presence of traffic lights. Although the general minimum cost path problem is NP-hard, both the computation

of minimum time paths and the computation of minimum cost paths in particular, but realistic indeed, street network scenarios, are shown to be efficiently solvable.

We think that further realistic scenarios could be investigated from an algorithmic and also from a computational point of view. This will be the subject of further research.

References

Ahuja, R.K., Magnanti T.L. and J.B. Orlin, Network flows: Theory, algorithms, and applications, Prentice Hall, Englewood Cliffs, NJ, 1993.

Desrochers, M., "An algorithm for the shortest path problem with resources constraints", Université de Montréal, Cahier du GERAD, G-88-27, 1988.

Desrochers, M. and F. Soumis, "A generalized permanent labelling algorithm for the shortest path problem with time windows", *INFOR* 26 (3), 191-212, (1988).

Desrosiers, J., Pelletier, P. and F. Soumis, "Plus court chemin avec contraintes d'horaires", *R.A.I.R.O.* 17 (4), 357-377, (1983).

Dial, R.B., "Algorithm 360: Shortest path forest with topological ordering", *Communications of the ACM* 12, 632-633, (1969).

Kirby, R.F. and R.B. Potts, "The minimum route problem for networks with turn penalties and prohibitions", *Transportation Research* 3, 397-408, (1969).

Orda, A. and R. Rom, "Shortest-path and minimum-delay algorithms in network with time-dependent edge length", *Journal of the ACM* 37 (3), 607-625, (1990).

Orda, A. and R. Rom, "Minimum weight paths in time-dependent network", *Networks* 21 (3), 295-320, (1991).

Pallottino, S. and M.G. Scutellà, "Shortest path algorithms in transportation models: classical and innovative aspects", *Equilibrium and Advanced Transportation Modelling*, P. Marcotte and S. Nguyen (Eds.), Kluwer Academic Publishers, 245-281, (1998).

Ziliaskopoulos, A. K., "Optimum path algorithms on multidimensional networks: Analysis, design, implementation and computational experience", Ph. D. Thesis, University of Texas at Austin, 1994.