Algorithmic Operations Research Vol.1 (2006) 94–103

Mathematical models of the delay constrained routing problem

Walid Ben-Ameur [a] Adam Ouorou [b]

[a]GET/INT - CNRS/SAMOVAR, Institut National des Télécommunications, 9 rue Charles Fourier, 91011 Evry, France.
[b]France Télécom - Division R&D , CORE/MCN, 38-40 rue du Général Leclerc, 92794 Issy-Les-Moulineaux cedex 9.

Abstract

Given a network with known link capacities and traffic demands, one can compute the paths to be used and the amount of traffic to be send through each path by solving a classical multi-flow problem. However, more quality of service constraints such as delay constraints, may be imposed and the routing problem becomes difficult to solve. We assume that the delay on each link depends on both its capacity and the total flow on it. We show that satisfying the delay constraints and the capacity constraints is an NP-complete problem. We give a convex relaxation of the delay constrained routing problem and present some ways to get upper and lower bounds on the problem.

Key words: routing problems in telecommunications, convex programming, delay.

1. Introduction

Given a traffic matrix and a network with known link capacities, one of the most classical network optimization problems consists in computing a multi-path-routing satisfying the capacity constraints. This problem is a multiflow problem solved by linear programming techniques [1,19]. Another classical problem consists in minimizing the mean end-to-end delay. The delay through a link l is generally computed on the basis of the classical M/M/1 formula : $\frac{1}{c_l - f_l}$ where c_l is the capacity of the link (in bytes per second), f_l is the traffic carried through the link (in bytes per second) [18]. Using this notation, $\frac{1}{c_l - f_l}$ is the time needed to carry one byte on link l. The total delay needed to carry a byte from a source s to a sink t using a path p is the sum of delays on the path links: $\sum_{l\in p} \frac{1}{c_l - f_l}$. One can easily show that minimizing the mean end-to-end delay is equivalent to minimizing $\sum_{l\in E} \frac{f_l}{f_l - c_l}$ where E is the set of all network links [8]. The constraints of the problem are the classical routing constraints: all commodities (traffic demands) should be routed through the network, and the flow on each link is always lower than its capacity. The problem described above has been studied by many authors (see, e.g., [8,20]). It is now well-handled as a convex problem: the cost function to be minimized is convex, and all the constraints are linear. It can also be solved using semidefinite programming (see, e.g., [23]).

Email: Walid Ben-Ameur [walid.benameur@int-evry.fr], Adam Ouorou [adam.ouorou@francetelecom.com].

Even if minimizing the mean end-to-end delay is useful, we think that more explicit constraints should be added to model practical problems. When we solve this problem, some paths with long delays may be used which can deteriorate the quality of service.

Modern communication networks will be based on different classes of service with different levels of quality of service. Some applications can accept long delays, but others may require very short delays. Said another way, we think that the right problem consists in computing a minimum cost routing satisfying the delay constraints of each demand in addition to capacity and demand constraints. A first attempt to solve similar problems is presented in [2]. Some heuristics have been proposed to minimize the number of used paths. Moreover, there is a lot of work on constrained routing in different contexts depending on the networking technology. However, it seems that only basic heuristics are provided and the delay on a link is generally assumed to be constant. Notice that our problem presentation is mainly based on communication networks but most of the material provided in this paper will also be useful in the context of transportation networks.

The aim of this paper is to show how delay constraints can be integrated. First, we define the delay constrained routing problem. Then, we study the theoretical complexity of this problem in Section 3.. Some convex relaxations are presented in Sections 4. to tackle the problem. The constraints considered in both relaxations are compared in Appendices B, C. Some algorithms are presented in Section 5. to compute upper bounds of the problem solution. We propose in Section 6. a convex optimization algorithm to solve the convex relaxation of the problem in order to get a lower bound.

In order to keep this paper reasonably short, we intend to provide numerical results elsewhere.

2. Notations

Let $G = (V, E)$ be the undirected graph representing the network. K denotes the set of commodities (traffic demands) where a commodity $k \in K$ has a source node $s(k)$, a sink node $t(k)$ and a positive value v_k (demand). We denote by $P(k) = \{P_k^1, ..., P_k^{|P(k)|}\}$ a set of paths of G connecting $s(k)$ to $t(k)$. Let α_k be an upper bound of the end-to-end delay that should be satisfied by all the paths that will be used to carry the commodity k. A real variable x_k^i related to P_k^i denotes the proportion of traffic carried through the path. A routing cost w_l is considered for each link l. It may depend on geographic distances and any other link characteristics.

The problem that we aim to solve can be written as follows:

$$\text{Minimize } \sum_{l \in E} w_l f_l \tag{1}$$

$$\sum_{i=1}^{i=|P(k)|} x_k^i \geq 1, \forall k \in K \tag{2}$$

$$\sum_{k \in K} \sum_{P_k^i \ni l} x_k^i v_k \leq f_l, \forall l \in E \tag{3}$$

$$f_l \leq c_l, \forall l \in E \tag{4}$$

$$\sum_{l \in P_k^i} \frac{1}{c_l - f_l} \leq \alpha_k, \forall k \in K, P_k^i \quad \text{such that } x_k^i > 0 \tag{5}$$

$$0 \leq x_k^i \leq 1, f_l \geq 0, \forall k \in K, P_k^i \in P(k), l \in E \tag{6}$$

The objective function (1) is the total routing cost. Constraints (2) are the classical demand constraints: all commodities should be satisfied. The second set of constraints (3) define the flow carried on each link. This flow should be lower than the capacity (constraints (4)). Constraints (6) express the fact that the flow variables are positive and that x_k^i are proportions of traffic. The delay constraints (5) express that the total delay through any path that is really used is lower than the maximum delay that can be accepted for commodity k. Notice that if a path P_k^i is not used ($x_k^i = 0$), there is no any delay constraint related to this path.

3. Complexity issues

When the set $P(k)$ of some demands contains an exponential number of paths, one can easily show that the delay constrained routing problem is NP-hard. This can be proved using a reduction from the multi-constrained shortest path [12,16] and the equivalence of separation and optimization. On the other hand, if only one path is considered for each demand, the delay constrained routing problem becomes a convex problem which is easy to solve.

A more interesting question to settle consists in studying the theoretical complexity when we consider more than one path for some demands and we assume that the sets $P(k)$ are already given (i.e., they are a part of the problem inputs).

We will show that the corresponding feasibility problem (described below) is NP-complete even if there are exactly two paths for each demand.

The feasibility problem consists in checking whether there is a routing solution satisfying the delay and the capacity constraints. We assume here that each set $P(k)$ contains exactly two paths.

Feasibility: is there a solution of the problem below

$$\sum_{i=1}^{i=|P(k)|} x_k^i \geq 1, \quad \forall k$$

$$\sum_{k \in K} \sum_{P_k^i \ni l} x_k^i v_k \leq f_l, \quad \forall l \in E$$

$$f_l \leq c_l, \quad \forall l \in E$$

$\sum_{l \in P_k^i} \frac{1}{c_l - f_l} \leq \alpha_k$, $\forall k \in K, P_k^i$ such that $x_k^i > 0$, $0 \leq x_k^i \leq 1$, $f_l \geq 0$, $\forall k \in K, P_k^i \in P(k), l \in E$

To prove the NP-completeness of the feasibility problem we will use a reduction from the set partition problem. Consider a set of positive integers $S = \{s_1, ..., s_n\}$ $(n > 1)$. The partition problem consists in checking wether it is possible to partition the set S in two subsets A and $S \setminus A$ such that $\sum_{s_i \in A} s_i = \sum_{s_i \in S \setminus A} s_i$. The partition problem is a well known NP-complete problem [12].

Let us build a network with $n + 4$ nodes in the following way. A node x_i is associated with each integer s_i. We also add four vertices a, b, z, t , and the following links: (x_i, a), (x_i, b), (a, t), (a, z), (b, t) and (b, z). We consider two commodities for each vertex x_i: a commodity d_i^t from x_i to t whose value is s_i and a second commodity d_i^z from x_i to z whose value is

$1+s_i$. The capacity of both links (x_i,a) and (x_i,b) is $1+s_i$. We also assume that the capacity of links (a,t) and (b,t) is $1+\frac{1}{2}\sum s_i$. Links (a,z) and (b,z) have an infinite capacity. Only two paths are allowed for each commodity. Commodity d_i^t can use the paths $x_i,\ a,\ t$ and $x_i,\ b,\ t$. Similarly, commodity d_i^z can only use paths $x_i,\ a,\ z$ and $x_i,\ ,b,\ z$. A delay constraint is defined for each commodity d_i^t: the delay on any used path should be at most equal to 2. No delay constraint is added for commodities d_i^z. Suppose that the routing problem related to this network is feasible. Both links (a,t) and (b,t) will be used to carry traffic and we necessarily have $f_{(a,t)}+f_{(b,t)}=\sum s_i$. At least one of the two links, let us say (a,t), satisfies $f_{(a,t)}\geq\frac{1}{2}\sum s_i=c_{(a,t)}-1$. This means that $\dfrac{1}{c_{(a,t)}-f_{(a,t)}}\geq 1$. Moreover, there is at least one commodity d_j^t using the path $x_j a t$, implying that $\dfrac{1}{c_{(a,t)}-f_{(a,t)}}+\dfrac{1}{c_{(x_j,a)}-f_{(x_j,a)}}\leq 2$. Combining the two previous inequalities leads to $\dfrac{1}{c_{(x_j,a)}-f_{(x_j,a)}}\leq 1$. Commodities d_j^t and d_j^z share the same links (x_j,a) and (x_j,b). Suppose that d_j^t is also using the path $x_j,\ b,\ t$. This implies that $f_{(x_j,b)}<c_{(x_j,b)}=1+s_j$, but we already assumed that $\dfrac{1}{c_{(x_j,a)}-f_{(x_j,a)}}\leq 1$ which leads to $f_{(x_j,b)}+f_{(x_j,a)}<2s_j+1$. Then, it becomes impossible to carry both demands d_j^t and d_j^z. Said another way, if d_j^t uses path $x_j,\ a,\ t$, then it does not use the other path and we necessarily have $\dfrac{1}{c_{(x_j,a)}-f_{(x_j,a)}}=1$. We should also have $\dfrac{1}{c_{(a,t)}-f_{(a,t)}}=1$. The same result is valid for link (b,t). In other terms, if the routing problem is feasible then $f_{(a,t)}=f_{(b,t)}=\dfrac{\sum s_i}{2}$ and each commodity d_i^t is single path routed. A partition of the set S is obtained by taking the set of commodities flowing through (a,t) and those using link (b,t). The other sense is trivial: having a partition, one can build a solution of the routing problem. As the reduction is polynomial, the feasibility problem is NP-complete.

Notice that the complexity of the feasibility problem associated with the delay constrained routing problem strongly depends on the expression of the delay $\dfrac{1}{c-f}$. If we assume, for example, that the delay on each link l is constant (does not depend on the flow), then the feasibility problem becomes polynomial. Indeed, as we assumed that the paths are a part of the problem inputs, we only have to keep those that satisfy the delay constraints and we get a convex problem that can be solved using any polynomial convex optimization algorithm.

4. A convex relaxation of the delay constrained routing problem

Constraints (5) are equivalent to the following constraints

$$x_k^i\left(\sum_{l\in P_k^i}\frac{1}{c_l-f_l}-\alpha_k\right)\leq 0,\qquad \forall k\in K,\ P_k^i\in P(k). \quad (7)$$

Since the mean delay problem stated in Section 1. is now efficiently solved as a convex problem, a good direction to get an efficient algorithm to solve our delay constrained routing problem consist in finding a convex model which can help in the solution of the problem.

All constraints (2, 3, 4) are linear constraints and thus convex. However, one can easily show that constraints (7) are not convex. We then propose to replace constraints (7) by the following convex constraints:

$$(x_k^i)^2\sum_{l\in P_k^i}\frac{1}{c_l-f_l}-x_k^i\alpha_k\leq 0, \quad (8)$$

Notice that this constraint is equivalent to the previous one when x_k^i is boolean. However, only constraint (8) is convex. This can be easily shown by computing the Hessian matrix of the left hand side function. As $x_k^i\alpha_k$ is linear, to show the convexity of the constraint (8), it is sufficient to prove that $g:(x_k^i,f_l)\to(x_k^i)^2\frac{1}{c_l-f_l}$ is convex (where $l\in P_k^i$). The Hessian matrix $\mathcal{H}$ of g is given below:

$$\mathcal{H}=\begin{pmatrix} 2\frac{1}{c_l-f_l} & 2x_k^i(\frac{1}{c_l-f_l})^2 \\ 2x_k^i(\frac{1}{c_l-f_l})^2 & (x_k^i)^2\frac{2}{(c_l-f_l)^3}\end{pmatrix}$$

$\mathcal{H}$ is clearly positive semidefinite.

If we use constraints (8) instead of constraints (5), we obtain a relaxation which is a convex problem that can be solved by convex programming techniques.

Another way to integrate the delay constraints (5) consists in adding the following constraints:

$$\sum_{l\in E}\frac{\sum_{P_k^i\ni l}x_k^i}{c_l-f_l}-\alpha_k\leq 0, \quad (9)$$

Constraint (9) dominates constraints (8). However, constraints (9) are also not convex. They become convex if we replace x_k^i by $(x_k^i)^2$. This leads to $\sum\limits_{l\in E} \frac{\sum\limits_{P_k^i \ni l} (x_k^i)^2}{c_l - f_l} - \alpha_k \leq 0$. They can be written in the following way, $\sum\limits_{P_k^i} (x_k^i)^2 \times \left(\sum\limits_{l\in P_k^i} \frac{1}{c_l - f_l} \right) - \alpha_k \leq 0$. The obtained inequalities are now dominated by constraints (8). Said another way, replacing x_k^i by $(x_k^i)^2$ does not seem to be very efficient. A more efficient way to transform constraints (9) consists in considering the square of $\sum\limits_{P_k^i \ni l} x_k^i$. Thus, we obtain the following inequalities:

$$\sum_{l\in E} \frac{(\sum\limits_{P_k^i \ni l} x_k^i)^2}{c_l - f_l} - \alpha_k \leq 0, \tag{10}$$

Constraints (10) are clearly valid for our delay constrained routing problem: $\sum\limits_{P_k^i \ni l} x_k^i \leq 1$ implies

$$\sum_{l\in E} \frac{(\sum\limits_{P_k^i \ni l} x_k^i)^2}{c_l - f_l} \leq \sum_{l\in E} \frac{\sum\limits_{P_k^i \ni l} x_k^i}{c_l - f_l}.$$

By combination with constraint (9), we obtain constraint (10).

One can easily compute the Hessian matrix to show that constraints (10) are convex. Notice that there is no any general domination relationship between constraints (10) and constraints (8). An example is given in Appendix B to show this fact.

A third set of convex constraints can be obtained using the fact that the delay on each link is necessarily upper bounded. Indeed, if a link l is not used, then $\frac{1}{c_l - f_l} \leq \frac{1}{c_l}$. Moreover, if l is used by at least one demand k, then $\frac{1}{c_l - f_l} \leq \alpha_k$. Combining these two observations, one can easily deduce an upper bound α_l (resp. α_k^i) for the delay on each link l (resp. path P_k^i). The upper bound α_k^i is assumed to be higher than α_k. More details on the methods that can be used to get these upper bounds are presented in Section 5.1..

Assuming that the upper bounds α_k^i are given, we introduce the following new set of inequalities:

$$\sum_{l\in P_k^i} \frac{1}{c_l - f_l} + x_k^i(\alpha_k^i - \alpha_k) - \alpha_k^i \leq 0 \tag{11}$$

The validity of these inequalities is obvious:

$$\sum_{l\in P_k^i} \frac{1}{c_l - f_l} = x_k^i \sum_{l\in P_k^i} \frac{1}{c_l - f_l} + (1 - x_k^i) \sum_{l\in P_k^i} \frac{1}{c_l - f_l} \leq x_k^i \alpha_k + (1 - x_k^i)\alpha_k^i$$

These constraints are clearly convex.

If we compare constraints (11) and constraints (8), one can easily see that constraints (8) dominate constraints (11) if and only if x_k^i is fractional and

$$x_k^i < \frac{\alpha_k^i}{\sum\limits_{l\in P_k^i} \frac{1}{c_l - f_l}} - 1.$$

This clearly shows that we should carefully compute the upper bounds α_k^i to obtain efficient constraints of type (11). Moreover, if α_k^i is sufficiently large, constraints (11) become very weak and do not dominate constraints (10). An example is given in Appendix (C) to show that constraints (11) are generally not dominated by constraints (10).

Combining all the results of this section leads to the following convex relaxation of the delay constrained routing problem. Notice that the positivity of the routing cost values w_l implies that $\sum\limits_{i=1}^{i=|P(k)|} x_k^i = 1$ for each demand k at optimality. The problem is rewritten as

$$\begin{aligned}
&\text{Minimize } \sum_{l\in E} w_l f_l \\
&\sum_{i=1}^{i=|P(k)|} x_k^i = 1,\ \forall k \in K, \\
&\sum_{k\in K} \sum_{P_k^i \in P(k): l \in P_k^i} x_k^i v_k = f_l,\ \forall l \in E, \\
&(x_k^i)^2 \sum_{l\in P_k^i} \frac{1}{c_l - f_l} - x_k^i \alpha_k \leq 0,\ \forall k \in K, P_k^i \in P(k) \\
&\sum_{l\in E} \frac{(\sum\limits_{P_k^i \ni l} x_k^i)^2}{c_l - f_l} - \alpha_k \leq 0,\ \forall k \in K, \\
&\sum_{l\in P_k^i} \frac{1}{c_l - f_l} + x_k^i(\alpha_k^i - \alpha_k) - \alpha_k^i \leq 0, \\
&\qquad\qquad \forall k \in K, P_k^i \in P(k), \\
&0 \leq x_k^i,\ \forall k \in K, P_k^i \in P(k), \\
&f_l \geq 0,\ f_l - c_l \leq 0,\ \forall l \in E.
\end{aligned} \tag{12}$$

5. Upper bounds of the delay constrained routing problem

This section is dedicated to upper bounds. We will first show how one can compute the numbers α_l and α_k^i introduced in Section 4.. Then, we provide a set of heuristics to compute an upper bound for the solution of the whole problem.

5.1. *On the upper bounds α_k^i and α_l*

Let us first focus on the upper bounds α_l of the delay $d_l = \frac{1}{c_l - f_l}$. We obviously have $d_l \geq \frac{1}{c_l}$. A first procedure that can be used to compute an upper bound α_{l_0} related to a link l_0 may consist in solving the following convex problem:

$$\text{Minimize } \sum_{l \in E} -f_{l_0}$$
$$-\sum_{i=1}^{i=|P(k)|} x_k^i + 1 \leq 0 \quad \forall k \in K$$
$$\sum_{k \in K} \sum_{P_k^i \ni l} x_k^i v_k - f_l \leq 0 \quad \forall l \in E$$
$$\sum_{k \in K} \sum_{P_k^i \ni l_0} -x_k^i v_k + f_{l_0} \leq 0$$
$$f_l - c_l \leq 0 \quad \forall l \in E$$
$$(x_k^i)^2 \sum_{l \in P_k^i} \frac{1}{c_l - f_l} - x_k^i \alpha_k \leq 0 \; \forall k \in K, P_k^i \in P(k)$$
$$\sum_{l \in E} \frac{(\sum_{P_k^i \ni l} x_k^i)^2}{c_l - f_l} - \alpha_k \leq 0 \quad \forall k \in K$$
$$0 \leq x_k^i \leq 1, f_l \geq 0 \quad \forall k \in K, P_k^i \in P(k), l \in E$$

One can easily see that the convex problem above is quite similar to the model of Section 4.. We only skip constraints (11) (because they use the upper bounds α_k^i) and we add the constraint $-\sum_{k \in K} \sum_{P_k^i \ni l_0} x_k^i v_k + f_{l_0} \leq 0$. The solution of the convex problem above will give us the bound $\alpha_{l_0} = \frac{1}{c_{l_0} - f_{l_0}}$.

Some other simple remarks can help to get another upper bound. Let $P(l)$ be the set of paths containing l: $P(l) = \{P_k^i \in P(k), P_k^i \ni l, k \in K\}$. If a path $P_k^i \in P(l_0)$ is used, then we should have $\frac{1}{c_{l_0} - f_{l_0}} + \sum_{l \in P_k^i, l \neq l_0} \frac{1}{c_l - f_l} \leq \alpha_k$. This implies that $\frac{1}{c_{l_0} - f_{l_0}} + \sum_{l \in P_k^i, l \neq l_0} \frac{1}{c_l} \leq \alpha_k$. In other terms, if $P_k^i \in P(l_0)$ is used, then $\frac{1}{c_{l_0} - f_{l_0}} \leq \alpha_k - \sum_{l \in P_k^i, l \neq l_0} \frac{1}{c_l}$.

A simple upper bound is then given by

$$\max\left(\frac{1}{c_{l_0}}, \max_{P_k^i \in P(l_0)} \alpha_k - \sum_{l \in P_k^i, l \neq l_0} \frac{1}{c_l}\right).$$

Notice that we can eliminate from the beginning all paths P_k^i such that $\sum_{l \in P_k^i} \frac{1}{c_l} > \alpha_k$ because these paths can never be used. This implies that the upper bound is simply

$$d_{l_0} \leq \max_{P_k^i \in P(l_0)} \alpha_k - \sum_{l \in P_k^i, l \neq l_0} \frac{1}{c_l}. \tag{13}$$

This bound can be slightly improved using the fact that the flow on a link l_0 is necessarily carried by the paths of $P(l_0)$. In fact, there is at least one path $P_k^i \in P(l_0)$ carrying at least $\frac{f_{l_0}}{|P(l_0)|}$. This implies that there is a path such that

$$\frac{1}{c_{l_0} - f_{l_0}} + \sum_{l \in P_k^i, l \neq l_0} \frac{1}{c_l - \frac{f_{l_0}}{|P(l_0)|}} \leq \alpha_k \tag{14}$$

It is easy to compute for each path $P_k^i \in P(l_0)$ the maximal value of f_{l_0} such that inequality (14) is satisfied. Let $f_{l_0}(P_k^i)$ denote this value. The flow on link l_0 will be lower than $\max_{P_k^i \in P(l_0)} f_{l_0}(P_k^i)$ and the new upper bound is

$$d_{l_0} \leq \frac{1}{c_{l_0} - \max_{P_k^i \in P(l_0)} f_{l_0}(P_k^i)}. \tag{15}$$

The upper bounds α_k^i will be simply based on the values of α_l computed as explained above. More precisely, we define α_k^i as follows:

$$\alpha_k^i = \max\left(\alpha^k, \sum_{l \in P_k^i} \alpha_l\right) \tag{16}$$

5.2. *Heuristics to compute upper bounds of the delay constrained routing problem*

A simple upper bound can be computed by adding the delay constraint $\sum_{l \in P_k^i} \frac{1}{c_l - f_l} \leq \alpha_k$ for each path. Said another way, even if the path is not used, we impose the constraint. We obtain a convex problem that can be solved by convex optimization. Notice that adding all these constraints may produce an infeasible problem leading to an infinite upper bound. This upper bound can be improved in some cases. When we solve the convex problem integrating a delay constraint for each path, the solution may not use all the paths. Then we can eliminate all the constraints related to the non used paths and solve again the new restricted convex problem without these path variables. This process can be repeated until all the paths considered in the master problem are effectively used to carry traffic.

$$\begin{aligned} \text{Minimize } & \sum_{l \in E} w_l f_l \\ & \sum_{i=1}^{i=|P(k)|} x_k^i \geq 1, \forall k \in K \\ & \sum_{k \in K} \sum_{P_k^i \ni l} x_k^i v_k \leq f_l, \forall l \in E \\ & f_l \leq c_l, \forall l \in E \\ & \sum_{l \in P_k^i} \frac{1}{c_l - f_l} \leq \alpha_k, \forall k \in K, P_k^i \in P(k) \\ & 0 \leq x_k^i \leq 1, f_l \geq 0, \forall k \in K, P_k^i \in P(k), l \in E \end{aligned}$$

In fact, any method to compute a feasible solution gives an upper bound. Many other heuristics can be proposed to compute a hopefully good solution. One can, for example, fix the delays d_l on the links and solve the corresponding routing problem using only the paths satisfying the delay constraints $d_k^i = \sum_{l \in P_k^i} \frac{1}{c_l - f_l} \leq \alpha_k$.

$$\begin{aligned} \text{Minimize } & \sum_{l \in E} w_l f_l \\ & \sum_{i=1}^{i=|P(k)|} x_k^i \geq 1, \forall k \in K \\ & \sum_{k \in K} \sum_{P_k^i \ni l, d_k^i \leq \alpha_k} x_k^i v_k \leq f_l, \forall l \in E \\ & f_l \leq c_l - \frac{1}{d_l}, \forall l \in E \\ & 0 \leq x_k^i \leq 1, x_k^i = 0 \text{ if } d_k^i > \alpha_k, \\ & \forall k \in K, P_k^i \in P(k) f_l \geq 0, \forall l \in E \end{aligned}$$

Then the values of d_l may be changed and the linear problem solved again to decrease the value of the objective function. One can implement several heuristics depending on how d_l values are changed. For instance, if we let $h(d_1, \ldots, d_l, \ldots, d_{|E|})$ be the optimal value of the above problem, we may look for $d_l,\ l \in E$ that minimize h. Notice that we already have an upper bound α_l for d_l. This upper bounds should be taken into account when the values of d_l are changed because there is no hope to find any feasible solution when $d_l > \alpha_l$. Let u_l denote the dual variable corresponding to the constraint $f_l \leq c_l - \frac{1}{d_l}$. If there is no current path P_k^i using the link l such that d_k^i is exactly equal to α_k, then a small variation of d_l will not change the set of paths that can be used for routing. Moreover, if we assume that the current linear program is non degenerate, then we know that $\frac{\partial h(d_1, \ldots, d_l, \ldots, d_{|E|})}{\partial d_l} = -\frac{u_l}{{d_l}^2}$. A simple subgradient descent method can be implemented to decrease the value of h.

Finally, consider the following convex multicommodity flow problem

$$\begin{aligned} \text{Minimize } & \sum_{l \in E} \log(\frac{1}{c_l - f_l}) \\ & \sum_{i=1}^{i=|P(k)|} x_k^i \geq 1, \forall k \in K \\ & \sum_{k \in K} \sum_{l \in P_k^i} x_k^i v_k \leq f_l, \quad \forall l \in E, \\ & 0 \leq f_l \leq c_l, \quad \forall l \in E \\ & 0 \leq x_k^i \leq 1, \quad \forall k \in K, P_k^i \in P(k). \end{aligned}$$

The Karush Kuhn Tucker conditions for this problem state that each commodity will be routed through the shortest paths in sense of the derivative of $\log(\frac{1}{c_l - f_l})$ (i.e., $\frac{1}{c_l - f_l}$). For each commodity k, let β_k be the common value of these shortest paths. If $\beta_k \leq \alpha_k$ for each k, the solution of the above problem provide an upper bound to the delay constrained routing problem. To have a chance to get a feasible solution, we can integrate the upper bounds α_l in the problem formulation. We only have to replace the constraints $f_l \leq c_l$ by the constraints $f_l \leq c_l - \frac{1}{\alpha_l}$. This is only a necessary condition to get a feasible solution.

6. Solution methods for lower bounds

The aim of the paper was to present some mathematical models of the delay constrained routing problem that can be handled by convex programming. An upper bound is obtained from the algorithms of section 5.2.. In this section, we sketch some algorithmic solutions for the convex problem (12) which provide a lower bound.

We think that the best way to deal with Problem (12) is to relax the constraints (8), (10) and (11), and solve the resulting dual problem. So, let $\gamma,\ \zeta,\ \eta$ be the dual variables associated respectively to these constraints. Then the standard Lagrangian function is

$$\begin{aligned} L(x,f,\gamma,\ \zeta,\ \eta) &= \sum_{l\in E} w_l f_l \\ &+ \sum_{k\in K}\sum_{P_k^i\in P(k)} \gamma_k^i((x_k^i)^2 \sum_{l\in P_k^i} \tfrac{1}{c_l-f_l} - \alpha_k x_k^i) \\ &+ \sum_{k\in K} \zeta_k(\sum_{l\in E} \tfrac{1}{c_l-f_l}(\sum_{P_k^i\in P(k):l\in P_k^i} x_k^i)^2 - \alpha_k) \\ &+ \sum_{k\in K}\sum_{P_k^i\in P(k)} \eta_k^i(\sum_{l\in P_k^i} \tfrac{1}{c_l-f_l} + (\alpha_k^i-\alpha_k)x_k^i - \alpha_k^i) \end{aligned}$$

Reordering terms, the Lagrangian function may be rewritten as

$$\begin{aligned} L(x,f,\gamma,\ \zeta,\ \eta) = \sum_{l\in E}[w_l f_l + \frac{1}{c_l-f_l} \\ (\sum_{k\in K}(\zeta_k(\sum_{P_k^i\in P(k):l\in P_k^i} x_k^i)^2 \\ + \sum_{P_k^i\in P(k):l\in P_k^i} (\gamma_k^i(x_k^i)^2 + \eta_k^i)))] \\ - \sum_{k\in K}[\zeta_k\alpha_k + \sum_{P_k^i\in P(k)} \\ (\gamma_k^i\alpha_k x_k^i + \eta_k^i\alpha_k^i - \eta_k^i(\alpha_k^i-\alpha_k)x_k^i)] \end{aligned}$$

The dual function is defined by

$$\mathcal{L}(\gamma,\ \zeta,\ \eta) = \min_{(x,f)\in X} L(x,f,\gamma,\ \zeta,\ \eta) \qquad (17)$$

where

$$\begin{aligned} X = \Big\{(x,f) : \sum_{i=1}^{|P(k)|} x_k^i = 1,\ \sum_{k\in K}\sum_{P_k^i\in P(k):l\in P_k^i} \\ x_k^i v_k = f_l,\ f_l - c_l \le 0,\ 0\le x_k^i, f_l \ge 0, \Big\} \end{aligned}$$

If we define $x(\gamma,\ \zeta,\ \eta), f(\gamma,\ \zeta,\ \eta)$ as the solution corresponding to the computation of $\mathcal{L}(\gamma,\ \zeta,\ \eta)$, it is well-known from convex analysis that $f_k^i\big(x(\gamma,\ \zeta,\ \eta), f(\gamma,\ \zeta,\ \eta)\big)$, $g_k\big(x(\gamma,\ \zeta,\ \eta), f(\gamma,\ \zeta,\ \eta)\big)$ and $h_k^i\big(x(\gamma,\ \zeta,\ \eta), f(\gamma,\ \zeta,\ \eta)\big)$ provide a subgradient to the $\mathcal{L}$ at $(\gamma,\ \zeta,\ \eta)$. The functions $f_k^i,\ g_k$ and h_k^i correspond to the constraints (8), (10) and (11) respectively.

For any $(\gamma,\ \zeta,\ \eta)$, $\mathcal{L}(\gamma,\ \zeta,\ \eta)$ is a lower bound to the optimal value of the problem (12) and hence a lower bound to problem (1)-(6).

We can remove the capacity constraints $f_l - c_l \le 0$ using a trick proposed in [8] by replacing the delay function $d(f_l) = 1/(c_l - f_l)$ by a function $\mathbf{d}(f_l)$ which is identical with d in $[0,\ \kappa c_l]$ and quadratic for $f_l > \kappa c_l$. The parameter κ is a real which must be chosen very close to 1, but appropriate choices can be obtained in the present context using the delay bound α_l computed in Section 5.1.. We can take $\kappa = 1 - \frac{1}{\alpha_l c_l}$. This quadratic function is chosen such that d and $\mathbf{d}$ has the same values, first and second derivatives at κc_l.

A possible algorithm for maximizing the dual function $\mathcal{L}$ is the cutting plane type algorithm. This kind of algorithm alternates between a master problem (using the linear approximation $\hat{\mathcal{L}}_t$ obtained from subgradients collected at different points $(\gamma^\tau,\ \zeta^\tau,\ \eta^\tau),\ \tau = 0,\dots,t$), and a subproblem (17). A solution of the later provides a lower bound Problem (1)-(6). We outline a general cutting plane algorithm as follows.

Algorithmic scheme

(1) Choose $\gamma^0,\ \zeta^0,\ \eta^0 \ge 0$ and set $t = 0$.
(2) Solve Subproblem (17) and denote by $x(\gamma^t,\ \zeta^t,\ \eta^t), f(\gamma^t,\ \zeta^t,\ \eta^t)$ the solution.
(3) Use $f_k^i(x(\gamma^t,\ \zeta^t,\ \eta^t), f(\gamma^t,\ \zeta^t,\ \eta^t)),\ g_k(x(\gamma^t,\ \zeta^t,\ \eta^t), f(\gamma^t,\ \zeta^t,\ \eta^t))$ and $h_k^i(\gamma^t,\ \zeta^t,\ \eta^t)$ to update the linear approximation $\hat{\mathcal{L}}$ and use it to get $(\gamma^{t+1},\ \zeta^{t+1},\ \eta^{t+1})$. Set $t = t+1$ and go to Step 2.

The basic issue is how to get the next iterate $(\gamma^{t+1},\ \zeta^{t+1},\ \eta^{t+1})$. The way in which these test points are generated using the linear approximation $\hat{\mathcal{L}}_t$ in Step 3 makes the difference among the cutting planes algorithms. Solving the current linear approximation

$$\max_{\gamma,\ \zeta,\ \eta\ge 0} \hat{\mathcal{L}}_t(\gamma,\ \zeta,\ \eta) \qquad (18)$$

and pick its optimal solution as the next test point, leads to Kelley's algorithm [17]. The use of the analytic centers of the polyhedrons made by the subgragients inequalities and the best upper bound obtained during the previous iterations yields analytic center cutting plane algorithm, see [13]. Bundle type algorithms [15] use the master problem

$$\max_{\gamma,\ \zeta,\ \eta\ge 0} \hat{\mathcal{L}}_t(\gamma,\ \zeta,\ \eta) - \frac{\lambda}{2}\|(\gamma,\ \zeta,\ \eta) - (\gamma^t,\ \zeta^t,\ \eta^t)\|^2$$

in place of (18), where λ is a positive parameter.

Anyway, a new proposal has to be transmitted to the subproblem (17) which will react to this test point with new revised cutting plane to improve the linear approximation. Now, assume that $\gamma,\ \zeta,\ \eta, \geq 0$ are fixed. The Lagrangian subproblem (17) is a multicommodity flow problem with a nonlinear differentiable convex cost which can be solved in various ways [20]. Let us consider the Frank-Wolfe algorithm because of its simplicity. At each iteration of this algorithm, a first-order approximation of the objective function is used to find a search direction and a line-search is performed in that direction to improve the current iterate. Note that, the objective function L in (17) can be seen as a function of path flows variable x_k^i while preserving its convexity (see Appendix A; however we still use f_l to denote $\sum\limits_{k\in K}\sum\limits_{P_k^i\in P(k):l\in P_k^i} x_k^i v_k$ in order to alleviate typewritting). The partial derivative of $L(x,\gamma,\ \zeta,\ \eta)$ w.r.t. (x_k^i) is given by

$$\begin{aligned}
c_k^i(x_k^i) &= \partial_{x_k^i} L(x,\gamma,\ \zeta,\ \eta) \\
&= v_k[\sum_{l\in E:l\in P_k^i} w_l + \sum_{l\in E}\mathbf{d}'(f_l) \\
&\quad (\sum_{p\in K}(\zeta_p(\sum_{P_p^i\in P(p):l\in P_p^i} x_p^i)^2 \\
&\quad + \sum_{P_p^i\in P(p):l\in P_p^i}(\gamma_p^i(x_p^i)^2+\eta_p^i)))] \\
&\quad +2\sum_{l\in E:l\in P_k^i}\mathbf{d}(f_l)(\gamma_k^i x_k^i + \zeta_k(\sum_{P_k^{i'}\in P(k):l\in P_k^{i'}} x_k^{i'})) \\
&\quad -\gamma_k^i\alpha_k + \eta_k^i(\alpha_k^i-\alpha_k).
\end{aligned}$$

The linearized subproblem (allowing to compute a search direction) at iteration t of the Frank-Wolfe algorithm is as follows.

$$\begin{aligned}
\min \sum_{k\in K}\sum_{P_k^i\in P(k)} & c_k^i((x_k^i)^t)x_k^i \\
\text{s.t.} \sum_{i=1}^{|P(k)|} & x_k^i = 1, \quad k\in K, \\
& x_k^i \geq 0, \quad k\in K, P_k^i\in P(k) \qquad (19)
\end{aligned}$$

where $c_k^i((x_k^i)^t)$ is the partial derivative at the current $(x_k^i)^t$. The above problem splits into K linear programs

$$\begin{aligned}
\min \sum_{P_k^i\in P(k)} & c_k^i((x_k^i)^t)x_k^i \\
\text{s.t.} \sum_{i=1}^{|P(k)|} & x_k^i = 1, \\
& x_k^i \geq 0, P_k^i\in P(k),
\end{aligned}$$

which can be solved by linear programming if we consider that the set of paths $P(k)$ are inputs of the problem, see Section 3..

7. Conclusion and further research

We defined the delay constrained routing problem where the delay is based on the classical M/M/1 waiting time formula. We showed that the corresponding feasibility problem is NP-complete even if we consider only two paths for each commodity. Then, we gave a convex relaxation to compute lower bounds. We also proposed some methods to compute upper bounds.

There are many possible extensions and research directions related to delay constrained routing.

First, we assumed that the path set is given for each demand, but it may be useful to generate paths using a kind of column generation algorithm. The convex relaxation should be adapted to integrate new paths and new constraints.

It is also possible to define a semidefinite relaxation of the delay constrained routing problem. More specialized algorithm should be used to solve the semidefinite program.

To let the paper reasonably short, we do not include computational experiments and postpone them to a subsequent paper. They are necessary to evaluate the quality of the gap we can obtain with our proposed methodology.

In addition to delay and capacity constraints, some other constraints are sometimes imposed. One can for example carry each commodity using only one path. Notice that single path routing has been studied by many authors (without delay constraints) [4,22]. Valid inequalities are generally added to improve the linear relaxation. When dealing with single path delay constrained routing, these inequalities can also be used to improve the convex relaxation.

Another kind of routing which is commonly used in Internet networks is shortest path routing (see, for example [5] and [11]). A weight is associated to each link and each commodity should be routed using the shortest paths in sense of these link weights. An interesting problem may consist in computing a set of weights such that the used shortest paths satisfy the delay and the capacity constraints.

In the context of modern communication networks, it becomes increasingly difficult to predict point-to-point traffic demands [6,21]. In fact, several services which require variable bandwidths, have to use the network, making traffic prediction very difficult. Since the delay on each link depends on the flow on the link, the end-to-end delay constraints may be unsatisfied when some traffic configurations are considered. A new approach proposed in [6] where traffic is modelled using a polyhedron instead of a matrix may be extended to integrate delay constraints.

In most of cases, the network should have some survivability features: if a link fails, then the network should be able to carry the traffic (see, for example, [3], [7], [10], [19]). Since the paths used during failures may be required to satisfy some delay constraints, the models presented in the papers should be modified to integrate this kind of constraints.

A Convexity of the objective function in (17)

We claim that the objective function in (17) expressed in term of path flows is convex. We show this fact here. Note that we can limit ourselves to the convexity of the left hand side function in (8) expressed in terms of path flows only, the proof is essentially the same for (9). Recall that we already proved that $(x_k^i)^2 \sum_{l \in P_k^i} \frac{1}{c_l - f_l} - x_k^i \alpha_k$ is a convex function when we consider the (x_k^i)'s and the f_l's as independent variables. Let x be the routing vector (whose components are the (x_k^i)'s) and let f be the flow vector (whose components are the f_l's). Let g any function of the form $(x_k^i)^2 \sum_{l \in P_k^i} \frac{1}{c_l - f_l} - x_k^i \alpha_k$. We know that $g(x, f)$ is convex. We aim to prove that $h(x) = g(x, f(x))$ is convex where we express f in terms of the components of x (i.e., $f_l = \sum_{k \in K} \sum_{P_k^i \in P(k): l \in P_k^i} x_k^i v_k$).

Given any number λ in $[0, 1]$, and any pair of vectors x and x', we have:

$$h(\lambda x + (1-\lambda)x') = g(\lambda x + (1-\lambda)x', f(\lambda x + (1-\lambda)x')).$$

But f is linear in x so $f(\lambda x + (1-\lambda)x') = \lambda f(x) + (1-\lambda) f(x')$. This leads to $h(\lambda x + (1-\lambda)x') = g(\lambda x + (1-\lambda)x', \lambda f(x) + (1-\lambda) f(x'))$.

By g convexity, we get

$$g(\lambda x + (1-\lambda)x', \lambda f(x) + (1-\lambda) f(x')) \leq \lambda g(x, f(x)) + (1-\lambda) g(x', f(x')).$$

Combining all the previous inequalities leads to

$$h(\lambda x + (1-\lambda)x') \leq \lambda h(x) + (1-\lambda) h(x').$$

B Comparison of constraints (8) and (10)

Let us consider a demand k with two paths P_k^1 and P_k^2 that have only one common edge l_0. We assume that P_k^1 (resp. P_k^2) contains another link l_1 (resp. l_2). Constraints (8) corresponding with the two paths are:

$$(x_k^1)^2 \left(\frac{1}{c_{l_0} - f_{l_0}} + \frac{1}{c_{l_1} - f_{l_1}} \right) \leq \alpha_k x_k^1,$$
$$(x_k^2)^2 \left(\frac{1}{c_{l_0} - f_{l_0}} + \frac{1}{c_{l_2} - f_{l_2}} \right) \leq \alpha_k x_k^2.$$

On the other hand, constraint (10) becomes:

$$(x_k^1)^2 \frac{1}{c_{l_1} - f_{l_1}} + (x_k^2)^2 \frac{1}{c_{l_2} - f_{l_2}} + (x_k^1 + x_k^2)^2 \frac{1}{c_{l_0} - f_{l_0}} \leq \alpha_k.$$

Moreover, we should have $x_k^1 + x_k^2 = 1$.

We can also choose our parameters such that

$$\alpha_k \left(\frac{1}{\frac{1}{c_{l_0} - f_{l_0}} + \frac{1}{c_{l_1} - f_{l_1}}} + \frac{1}{\frac{1}{c_{l_0} - f_{l_0}} + \frac{1}{c_{l_2} - f_{l_2}}} \right) = 1.$$

In this case, it is easy to see that the solution

$$x_k^1 = \alpha_k \frac{1}{\frac{1}{c_{l_0} - f_{l_0}} + \frac{1}{c_{l_1} - f_{l_1}}} \quad \text{and} \quad x_k^2 = \alpha_k \frac{1}{\frac{1}{c_{l_0} - f_{l_0}} + \frac{1}{c_{l_2} - f_{l_2}}},$$

satisfies constraints (8), although

$$(x_k^1)^2 \frac{1}{c_{l_1} - f_{l_1}} + (x_k^2)^2 \frac{1}{c_{l_2} - f_{l_2}} + (x_k^1 + x_k^2)^2 \frac{1}{c_{l_0} - f_{l_0}} = \alpha_k + 2 x_k^1 x_k^2 \frac{1}{c_{l_0} - f_{l_0}} > \alpha_k.$$

In other words, constraints (10) are not dominated by constraints (8).

To show the opposite, we can assume that $\alpha_k = 11\frac{1}{c_{l_0}-f_{l_0}}$ and $\frac{1}{c_{l_1}-f_{l_1}} = \frac{1}{c_{l_2}-f_{l_2}} = 15\frac{1}{c_{l_0}-f_{l_0}}$. Constraints (8) become $x_k^1 \leq \frac{11}{16}$ and $x_k^2 \leq \frac{11}{16}$. Constraint (10) can be written as follows: $(x_k^1)^2 + (x_k^2)^2 \leq \frac{2}{3}$. Simple calculation show that $x_k^1 = \frac{1+\sqrt{\frac{1}{3}}}{2}$ and $x_k^2 = \frac{1-\sqrt{\frac{1}{3}}}{2}$ satisfy constraint (10) while the constraint $x_k^1 \leq \frac{11}{16}$ is violated. Hence, constraints (8) are not dominated by constraints (10).

C Comparison of constraints (11) and (10)

We already mentioned in Section 4. that constraints (11) do not dominate constraints (10) at least when the upper bounds α_k^i are very large. We should show here that constraints (11) are not dominated by constraints (10). We use the same example of Appendix B. We assume again that $\alpha_k = 11\frac{1}{c_{l_0}-f_{l_0}}$ and $\frac{1}{c_{l_1}-f_{l_1}} = \frac{1}{c_{l_2}-f_{l_2}} = 15\frac{1}{c_{l_0}-f_{l_0}}$. We also suppose that $\alpha_k^1 = \alpha_k^2 = 17\frac{1}{c_{l_0}-f_{l_0}}$. Constraints (11) become $x_k^1 \leq \frac{1}{6}$ and $x_k^2 \leq \frac{1}{6}$. The solution $x_k^1 = \frac{1+\sqrt{\frac{1}{3}}}{2}$ and $x_k^2 = \frac{1-\sqrt{\frac{1}{3}}}{2}$ satisfy constraint (10) while the constraint $x_k^1 \leq \frac{1}{6}$ is violated. Hence, constraints (11) are not dominated by constraints (10).

References

[1] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, *Network flows*, Prentice-Hall, 1993.

[2] S. Beker, D. Kofman and N. Puech, *Off-line MPLS layout design and reconfiguration: reducing complexity under traffic conditions*, Proceedings of the International Network Optimization Conference, Evry-Paris, 2003, pp. 61-66.

[3] W. Ben-Ameur, *Constrained length connectivity and survivable networks*, Networks 36 (2000) , 17-33.

[4] W. Ben-Ameur and Meriema Belaidouni, *A Superadditive Approach to Solve the Minimum Cost Single Path Routing Problem Preliminary Results*, Proceedings of the International Network Optimization Conference, Evry-Paris, 2003, pp. 67-71.

[5] W. Ben-Ameur and E. Gourdin, *Internet Routing and Related Topology Issues*, SIAM J. Discrete Mathematics, 17 (2003), pp. 18-49.

Received 17 November 2004; revised 22 April 2005; accepted 28 May 2005

[6] W. Ben-Ameur and Hervé Kerivin, *New economical virtual private networks*, Communications of the ACM, 46 (2003), pp. 69-73.

[7] W. Ben-Ameur and Loan Pham, *Design of survivable networks based on end-to-end rerouting*, Proceeding of DRCN20001, Budapest 2001.

[8] D.P. Bertsekas and R.G. Gallager, *Data Networks*, Englewood Cliffs, NJ: Prentice-Hall, 1987.

[9] D.P. Bertsekas, *Nonlinear Programming*, Athena Scientific: Belmont, MA, 1995.

[10] G. Dahl, and M. Stoer, *A cutting plane algorithm for multicommodity survivable network design problems*, INFORMS Journal on Computing, 10 (1998), pp. 1-11.

[11] B. Fortz and M. Thorup, *Internet Traffic Engineering by Optimizing OSPF Weights*, Proceedings of IEEE INFOCOM, 2000, pp. 519-528.

[12] M. Garey and D.S. Jhonson, *Computers and Intractability*, W.H. Freeman and Company, 1979.

[13] J.-L. Goffin, J.-Ph. Vial, *Convex nondifferentiable optimization: a survey focused on the analytic center cutting plane method*, Optimization Methods and Software 17(5), pp. 805-867, 2002.

[14] M. Grötschel, L. Lovász and A. Schrijver, *Geometric algorithms and combinatorial optimization*, Springer-Verlag, Berlin, 1988.

[15] J.-B. Hiriart-Urruty, C. Lemaréchal, *Convex Analysis and Minimization Algorithms*, Springer-Verlag, Berlin, 1993.

[16] J.M. Jaffe, *Algorithms for finding paths with multiple constraints*, Networks, 14 (1984), pp. 95-116.

[17] J. E. Kelly, *The cutting-plane method for solving convex programs*, Journal of the SIAM, 8 (1960), pp. 703–712.

[18] L. Kleinrock, *Theory, Volume 1, Queueing Systems*, John Wiley & Sons, New York, USA, 1975.

[19] M. Minoux, *Network Synthesis and Optimum Network Design Problems: Models, Solution Methods and Applications*, Networks, 19 (1989), pp. 313-360.

[20] A. Ouorou, P. Mahey, and J.P. Vial, *A survey of algorithms for convex multicommodity flow problems*, Management Science, 46 (2000), pp. 126-147.

[21] A. Ouorou, *Robust capacity assignment in telecommunications*, Computational Management Science, to appear.

[22] K. Park, S. Kang and S. Park *An Integer Programming Approach to the Bandwidth Packing Problem*, Management Science, 42 (1996), pp. 1277-1291.

[23] C. Touati, E. Altman, and J. Galtier, *Semi-definite programming approach for bandwidth allocation and routing in networks*, Game Theory and Applications, 9 (2003), pp. 169-179.