INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

IFORS International Federation of Operational Research Societies

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 17 (2010) 103–117
DOI: 10.1111/j.1475-3995.2009.00698.x

Sensitivity analysis on the all pairs q-route flows in a network

Madiagne Diallo[a], Serigne Gueye[b] and Pascal Berthomé[c]

[a]*Engenharia Industrial, Pontifícia Universidade Católica (PUC-Rio), Rua Marquês de São Vicente, Gávea Rio de Janeiro, RJ 22453-900, Brazil,*

[b]*Université du Havre, Laboratoire de Mathématiques Appliquées du Havre, 25 rue Philippe Lebon, BP 540, 76058 Le Havre cedex, France,*

[c]*Laboratoire d'Informatique Fondamentale d'Orléans (LIFO), Ecole Nationale Supérieure d'Ingénieurs de Bourges, 88 bld Lahitolle, 18020 Bourges cedex, France*

E-mail: diallo@puc-rio.br [Diallo]; serigne.gueye@univ-lehavre.fr [Gueye]; pascal.berthome@ensi-bourges.fr [Berthomé]

Received 14 August 2008; received in revised form 18 January 2009; accepted 18 January 2009

Abstract

Given an undirected edge-weighted network in which one edge capacity is allowed to vary, we propose in this paper a polynomial algorithm that needs only two 2-cut-trees computations to provide the all pairs maximum 2-route flow values. Moreover, we provide an extension to the case where k edge capacities may vary, and show that 2^k 2-cut-trees computations are necessary in order to determine the all pairs maximum 2-route-flow values. This makes our algorithm polynomial whenever $k = O(poly \log n)$.

Keywords: multiroute flows; q-route flows; 2-cut-tree; parametric flows; sensitivity analysis; cut-tree

1. Introduction

In this paper, we are concerned with the analysis of the impact of edge-capacity variation on the flows through a network. Each flow, called q-route flow, is supposed to use a fixed number $q > 0$ of weighted routes.

The relevance of using q-route flows is based on the fact that computer networks need some resiliency in order to insure information's propagation. In other words, they need to be fault tolerant meaning that a failure does not prevent all information from arriving at destination. As for real-life applications, one may cite Multi-Protocol Label Switched networks in which routing with service restorability is of much relevance, or even optical networks where it is a necessity. As for restoration, each connection must have an active path and a disjoint back-up path. The latter enables service restoration upon active path failure.

Published by Blackwell Publishing, 9600 Garsington Road, Oxford, OX4 2DQ, UK and 350 Main St, Malden, MA 02148, USA.

Let $G = (N, E, c)$ be a weighted connected undirected network, where $N = \{1, 2, \ldots, n\}$ is the set of nodes, E is the set of edges and c is the non-negative edge-capacity function. Let us consider in G a source–sink pair (s, t) of distinct nodes in N. An "elementary q-route set" from s to t in G is defined as a set of q edge-disjoint routes. When a flow is sent from s to t, to each respective q-route set is assigned a non-negative weight that represents the amount of flow going through each of its q edge-disjoint routes. A q-route flow from s to t is an assignment of the q-route sets from s to t and the respective non-negative weights. Thus, the value of a q-route flow is equal to the sum of the weights assigned to its elementary q-route sets. For each edge $e \in E$, the corresponding flow value on e (i.e., the sum of weights of routes that use e) is no more than its capacity $c(e)$.

If $q = 1$, computing the maximum q-route flow value corresponds to solving the classical max-flow problem popularized and solved by Ford and Fulkerson (1973). In the sequel, we will use the expression classical max-flow to designate flow as in the Ford and Fulkerson setting.

Using q-route flows, in the case of $q - 1$ edge failures, one may guarantee that the arriving flow at destination will not be null. Indeed, considering an elementary q-route set, if $q - 1$ edges fail, there remains at least one connection between the source and the destination.

Example. Consider the network of Fig. 1.

Case $q = 1$: R_1 and R_2 are elementary 1-route sets of the given network:

$$\begin{array}{lllllll} R_1: & s & \rightarrow & 2 & \rightarrow & t & \\ R_2: & s & \rightarrow & 1 & \rightarrow & 2 & \rightarrow & t \end{array}$$

In this case, an elementary q-route set contains only one route from s to t. Then, assigning the weight 2 to R_1 and 1 to R_2 gives a 1-route flow from s to t with value 3 that corresponds, as mentioned later, to the classical max-flow value.

Case $q = 2$: Consider the elementary 2-route sets R_3 and R_4 in the network:

$$R_3: \begin{cases} s \rightarrow 2 \rightarrow t \\ s \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow t \end{cases}; \quad R_4: \begin{cases} s \rightarrow 2 \rightarrow 3 \rightarrow t \\ s \rightarrow 1 \rightarrow 2 \rightarrow t \end{cases}$$

Assigning to R_3 the weight 1/2 and to R_4 the weight 1/2 gives a 2-route flow with value 1.

Now suppose a failure of link $(2, t)$: in Case 1, since both routes in R_1 and R_2 use the arc $(2, t)$, the incoming flow in t will be reduced to the value 0. If the flow represented some information, it would not arrive at destination. On the other hand, in Case 2, each 2-route set guarantees one connection aggregating flow to the value 1.

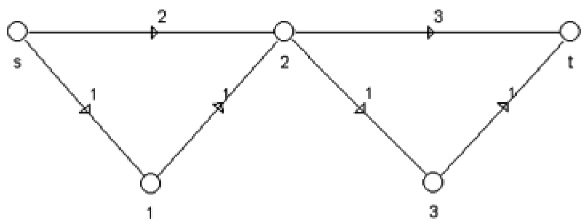

Fig. 1. Simple flow network.

The concept of q-route flow was introduced and studied by Kishimoto (1996). Later, subsequent results have been provided by Arggawal and Orlin (2002), Du (2003) and Kabadi et al. (2005). Kishimoto (1996) has proved that the well-known max-flow/min-cut theorem can be extended to the context of q-route flow. Such a result pointed out questions with respect to the extension of Gomory and Hu (1961) in the setting of q-route flow. Given an undirected network with n nodes, a cut-tree is a particular weighted tree obtained with the resolution of only $n-1$ classical max-flow/min-cut problems in order to determine the $\frac{n(n-1)}{2}$ classical max-flow/min-cut values between all pairs of nodes in the network. The particularity of a cut-tree remains in the fact that, the removal of each edge of the tree provides a minimum cut between the removed edge nodes, and the removed edge weight is the value of the provided minimum cut. Between any two nodes, the maximum value is the minimum weight on the unique path in the tree between the two nodes. It has been shown in Kabadi et al. (2005) that in the setting of q-route flow, a cut-tree, now called q-cut-tree, may be defined and computed polynomially only for $q = 2$, where a cut-tree is also a 2-cut-tree (Kabadi et al., 2005). The authors illustrated with a counter-example, that for $q \geqslant 3$ a q-cut-tree does not exist. The authors have also showed that a 2-cut-tree is not always a cut-tree.

All these later works dealt with constant edge capacities. Now let us suppose that in the considered network one or more edge capacities may vary. For short, we will call these edge capacities "parameterized capacities." In this setting, the interesting question was to compute efficiently the all pairs maximum flow values in undirected networks. The authors in Berthomé et al. (2003) and Barth et al. (2006) revisited the problem and proposed solutions as improvements and generalization of the work in Elmaghraby (1964). The authors showed that if there are k edges with parameterized capacities in the network, 2^k cut-trees are enough to solve in time $2^k O(n)$ the sensitivity analysis problem. Their method is well suited when k is small.

Our purpose in this paper is to extend these results to the 2-route flow case. Using the results of Kishimoto (1996) and Kabadi et al. (2005) highlighted above, we derive new consequences showing that if there are k edges with parameterized capacities, also 2^k 2-cut-trees are enough to determine the maximum 2-route flow value between any pair of source and destination for any assignment of the edge capacities.

Because of its theoretical and practical interests in fault-tolerant network design, the multiroute flow problem has been actively addressed in several references. Important results from these references to point out are the following. Kishimoto (1997) has defined a δ-reliable flow ($\delta \in]0, 1]$) as a flow for a given source–sink pair of nodes in a network where no edge carries a flow more than a fraction δ of the total flow in the network. In fact, if $\frac{1}{\delta}$ is an integer, thus taking $q = \frac{1}{\delta}$, q-route flows and a δ-reliable flow are linked by the following theorem (see Kishimoto, 1996; Aneja et al., 2007): *there exists a δ-reliable flow of value $\frac{\lambda}{\delta}$ if and only if there exists a q-route flow in G of value λ.* This theorem has been extended in Aneja et al. (2007) even in the case where $q = \frac{1}{\delta}$ is not an integer, and applied on a wavelength division multiplexing problem. Many algorithms to find maximum q-route flow value have also been proposed (see Kishimoto, 1996; Arggawal and Orlin, 2002; Du and Chandrasekaran, 2006; Du and Kabadi, 2005). All of them consider in a first step a q-route arc flow linear formulation of the problem. Then solving this formulation, the flows on q-route paths are derived in a second step. Kishimoto (1996) was the first to propose an algorithm to obtain q-route path flows from q-route arc flows. The complexity of Kishimoto's algorithm is $O(|E|^3 T_1)$ where T_1 is the time to find a perfect match in a bipartite graph. Arggawal and Orlin (2002) improve this complexity to $O(|E|T_2)$ where T_2 is the time to solve a unit-capacity maximum 1-route flow problem. Finally, Du and Kabadi (2005) propose an $O(|E|^2)$ algorithm, which is, to our knowledge, the best result in the current state of the art. Du and Chandrasekaran (2006) have shown how Kishimoto's and Aggarwal and Orlin's algorithms can be unified in a common framework based on Newton's method. In the same paper (Du and Chandrasekaran, 2006), the notion of q-path has been introduced, leading to an augmenting-path algorithm for the maximum q-route flow problem that extends Ford and Fulkerson's augmenting-path algorithm for the classical max-flow problem.

The remainder of the paper is organized as follows. In Section 2, some basic definitions and results on 2-route flows will be recalled. In Section 3, we will examine the case where only one edge has a parameterized capacity. The results reported in this section are then generalized to the case of several (k) parameterized capacities in Section 4. The paper ends with concluding remarks.

2. Preliminaries

Here are recalled definitions of q-capacity of a cut, q-minimum cut, q-cut-tree, and cuts of types I and II (Kabadi et al., 2005). Known theoretical results on q-route flows are also provided without proofs.

Definition 2.1 (q-capacity). *Let $G = (N, E, c)$ be a weighted undirected network. For any cut $(X, \bar{X})$ in G, let the edges in the cut be $\{e_1, e_2, \ldots, e_p\}$ arranged in decreasing order of their capacities, i.e. $c(e_1) \geqslant c(e_2) \geqslant \ldots \geqslant c(e_p)$. For any $j \in \{1, 2, \ldots, p\}$, we denote $\alpha_j(X)$ as*

$$\alpha_j(X) = \sum_{i=j}^{p} c(e_i).$$

The q-capacity of the cut $(X, \bar{X})$ is

$$\beta_q(X) = \min\left\{\frac{1}{q-j+1}\alpha_j(X) | 1 \leqslant j \leqslant q\right\}.$$

Thus, for $q = 2$, the 2-capacity of a cut is simply defined as the minimum of $\frac{1}{2}\alpha_1(X)$ and $\alpha_2(X)$. A cut is said to be of *type I* if $\beta_2(X) = \frac{1}{2}\alpha_1(X)$ and of *type II* if $\beta_2(X) = \alpha_2(X) < \frac{1}{2}\alpha_1(X)$. As for $q = 1$, one verifies that $\beta_1(X) = \alpha_1(X)$ corresponds to the capacity of a minimum cut in the classical max-flow problem.

Definition 2.2 (q-minimum cut). *Let (s, t) be a source–sink pair of distinct nodes in N. A cut $(S, \bar{S})$ (i.e., $S \subseteq N - \{t\}, s \in S, \bar{S} = N - S$) is said to be a q-minimum cut separating s and t in G iff*

$$\beta_q(S) = \min_{\substack{X \subseteq N - \{t\} \\ s \in X}} \beta_q(X).$$

The following theorem illustrates that as in the classical max-flow, the maximum value of a q-route flow and the q-capacity of a q-minimum cut are related.

Theorem 2.3 (Kishimoto, 1996). *For any network $G = [N, E, c]$ and a source–sink pair (s, t) of distinct nodes in N, the maximum value of a q-route flow from s to t in G is equal to the q-capacity of a q-minimum cut separating s and t in G.*

Hence, for the analysis of maximum q-route flows values, it is sufficient to look at the behavior of the q-capacities of q-minimum cuts. In the case $q = 2$, we know that the all pairs maximum 2-route flows values (i.e., 2-capacities of 2-minimum cuts) can be obtained from a 2-cut-tree defined as follows.

Definition 2.4 (2-cut-tree Kabadi et al., 2005). *Given an undirected network $G = (N, E, c)$, a weighted tree $T = (N, F, c_T)$ is said to be a 2-cut-tree of G iff for each edge $e = [i,j] \in F$, the fundamental cut (N_i^e, N_j^e) of T (i.e., the two node sets created by the deletion of e from T with $i \in N_i^e$ and $j \in N_j^e$) is a 2-minimum cut separating i and j and $c_T(e) = \beta_2(N_i^e)$.*

This q-cut-tree extends the Gomory–Hu cut-tree defined for the classical max-flow problem. In the following, the 1-cut-tree will be referred to as the classical cut-tree. The following results illustrates that cut-tree properties are maintained in a q-cut tree.

Lemma 2.5 (Kabadi et al., 2005). *Let $T = (N, F, c_T)$ be a 2-cut-tree of a network G. For any pair of distinct nodes (x, y) in N, if $e = [i,j] \in F$ is an edge with the smallest weight $c_T(e)$ on the unique path joining x and y in tree T, then the fundamental cut (N_i^e, N_j^e) is a 2-minimum cut separating x and y in network G.*

Hence, by this lemma, the 2-capacities of 2-minimum cuts (i.e., maximum 2-route flows values) between all pairs of nodes in G are resumed as edge weights in the 2-cut-tree. In addition, to obtain this 2-cut tree one can use the following theorem.

Theorem 2.6 (Kabadi et al., 2005). *If $T = (N, F, c_T)$ is a 1-cut-tree of a network G, then $T' = (N, F, c_{T'})$ is a 2-cut-tree of G, where $c_{T'}$ are the 2-capacities of the fundamental cuts of T.*

Given (s, t) a pair of nodes of a G, it is important to note that the 2-minimum and the 1-minimum cuts may be distinct sets. Indeed, let us consider the unique path in T or T' joining s and t, and suppose that the length of this path is at least 2. Let $e \in F$ (resp. e') be the edge with the lowest weight 1-capacity (resp. 2-capacity) in this path. Since c_T differs from c_T the edges e and e' may also differ. Consequently, the corresponding 1-minimum cut (N_i^e, N_j^e), and 2-minimum cut $(N_i^{e'}, N_j^{e'})$ may be distinct. Obviously, in the case where the size of the path is 1, $(N_i^e, N_j^e) = (N_i^{e'}, N_j^{e'})$.

3. Sensitivity analysis for one parameterized capacity

Let us suppose that in G the capacity of one edge $e = [i,j]$ is a parameter as $\lambda \geqslant 0$ (i.e., $c(e) = \lambda$) and let the resulting graph be denoted G^λ. Let (s, t) be a source–sink pair of distinct nodes of G^λ.

We denote by $f^2_{s,t}(\lambda)$ the maximum 2-route flow value between s and t in G^λ. For any cut $(X, \bar{X})$, containing e in G^λ, the 2-capacity of the cut will be denoted by $\beta^\lambda_2(X)$, where α^λ_1 and α^λ_2 will be the values involved in $\beta^\lambda_2(X)$ computation, i.e.

$$\beta^\lambda_2(X) = \min\left\{\frac{1}{2}\alpha^\lambda_1(X), \alpha^\lambda_2(X)\right\}.$$

The goal of the network sensitivity analysis is to compute efficiently all values $f^2_{s,t}(\lambda)$ for all node pairs (s,t) of G^λ. Let $\mathscr{C}$ be the set of cuts separating s and t in G, $\mathscr{C}_e$ its component of cuts that contain e and $\bar{\mathscr{C}}_e = \mathscr{C} - \mathscr{C}_e$ the complement of cuts of $\bar{\mathscr{C}}_e$ in $\mathscr{C}$ (i.e., set of cuts that do not contain e).

Note that these sets remain constant whatever the parameterized capacity is (for $\lambda = 0$, we consider that edge e exists, using limit argument). With these notations, it can be seen that

$$\forall\lambda \geqslant 0,\ f^2_{s,t}(\lambda) = \min\left\{\min_{C\in C_e}\beta^\lambda_2(C), \min_{C\in\bar{C}_e}\beta^\lambda_2(C)\right\}. \tag{1}$$

Since the parameterized edge does not belong to a cut in $\bar{\mathscr{C}}_e$, the quantity $\min\limits_{C\in\bar{C}_e}\beta^\lambda_2(C)$ remains constant under the capacity variation.

The exact value of this quantity may be derived with the lemmata below.

Lemma 3.1. *Let C be a cut containing the parameterized edge (i.e. $C \in \mathscr{C}_e$). We have*

$$\beta^\lambda_2(C) = \min\left\{\alpha^0_1(C), \frac{1}{2}\alpha^0_1(C) + \frac{1}{2}\lambda, \alpha^0_2(C) + \lambda\right\}. \tag{2}$$

Proof. Using the definition of the 2-capacity of a cut, we have

$$\beta^\lambda_2(C) = \min\left\{\frac{1}{2}\alpha^\lambda_1(C), \alpha^\lambda_2(C)\right\}.$$

Since e belongs to the cut, the second term clearly behaves as

$$\alpha^\lambda_1(C) = \alpha^0_1(C) + \lambda.$$

Furthermore, considering the α^λ_2 term, for $\lambda \leqslant c^0_1(C)$ (where, $c^0_1(C)$ is the greatest value among edge capacities in C), the parameterized capacity must be entirely considered. As for $\lambda > c^0_1(C)$, the parameterized capacity is the greatest one in the cut, and, thus, should not be considered in α^λ_2, but $c^0_1(C)$ should be included. Given that $\alpha^0_1(C) = \alpha^0_2(C) + c^0_1(C)$, we have

$$\alpha^\lambda_2(C) = \min\{\alpha^0_2(C) + \lambda, \alpha^0_1(C)\}.$$

Using both terms, we conclude the proof. ■

Corollary 3.2. *Given any s and t, there exists a finite value, denoted $f^2_{s,t}(\infty)$, that the 2-route flow value $f^2_{s,t}(\lambda)$ reaches at some constant $M \geqslant 0$ [independent of the source–sink pair (s,t)] and then stagnates.*

$$i.e.,\ \exists M \geqslant 0\ \forall\lambda_1, \lambda_2 \geqslant M | f^2_{s,t}(\lambda_1) = f^2_{s,t}(\lambda_2) = f^2_{s,t}(\infty).$$

Proof. Using Equation (1) and Lemma 3.1, the 2-route flow value is the minimum of a finite collection of functions of type Equation (2) and of some constant functions. Thus, for any s and t, there exists a value $M_{s,t}$ beyond which the 2-route flow value function is constant.

Let $M = \underset{s,t}{Max}\{M_{s,t}\}$, we have

$$\forall \lambda_1, \lambda_2 \geqslant M | f^2_{s,t}(\lambda_1) = f^2_{s,t}(\lambda_2) = f^2_{s,t}(\infty). \quad \blacksquare$$

With this corollary, we can then obtain the exact value of

$$\min_{C \in \bar{C}_e} \beta_2^{\lambda}(C)$$

Lemma 3.3. *Using the previous definitions, if s and t are not extremities of e, we have*

$$\min_{C \in \bar{C}_e} \beta_2^{\lambda}(C) = f^2_{s,t}(\infty).$$

Proof. From Corollary 3.2, there exists a value M for which any 1-cut-tree (at M) also defines a 1- and 2-cut tree for any $\lambda > M$. Let T_M be such a 1-cut-tree and $e = [i,j]$. Recall that any edge of T_M represents a fundamental cut (1-minimum cut or 2-minimum cut) (Gomory and Hu, 1961; Kabadi et al., 2005) in the network between s and t. Since M is enough large, i and j are neighbors in T_M. Since s and t are not extremities of e, no fundamental cut between s and t can contain e. Thus, we have the desired property. $\blacksquare$

In the case $e = [i,j] = [s,t]$, $\bar{\mathscr{C}}_e = \emptyset$, and, Equation (1) simplifies into

$$\forall \lambda \geqslant 0,\ f^2_{s,t}(\lambda) = \min_{C \in C_e} \beta_2^{\lambda}(C). \tag{3}$$

Then, using Lemma 3.3, if s and t are not extremities of e, Equation (1) becomes

$$\forall \lambda \geqslant 0\ f^2_{s,t}(\lambda) = \min\{\min_{C \in C_e} \beta_2^{\lambda}(C), f^2_{s,t}(\infty)\}. \tag{4}$$

Note that this equation remains valid even if $e = [s,t]$. Thus, in order to complete the computation of $f^2_{s,t}(\lambda)$, it remains to find the value $\min_{C \in C_e} \beta_2^{\lambda}(C)$.

Now, let $(X_2, \bar{X}_2)$ be a 2-minimum cut separating s and t in G^0. We suppose that $e \in (X_2, \bar{X}_2)$. Let us define $(X_1, \bar{X}_1)$ as follows

$$(X_1, \bar{X}_1) = \arg\min_{C \in C_e} \beta_1^0(C)$$

i.e., a cut of minimum 1-capacity in the set $\mathscr{C}_e$.

These cuts can be easily obtained with a 1-cut-tree and a 2-cut-tree of G^0.

In the following, for easier notation, in any cut $(Y, \bar{Y}) \in C$, $\alpha_1(Y)$ and $\alpha_2(Y)$ will denote the values computed in G^0 (i.e., $\alpha_1(Y) = \alpha_1^0(Y), \alpha_2(Y) = \alpha_2^0(Y)$). Also, saying $(Y, \bar{Y}) \in C$ is of type I or II refers to computations made in G^0.

The following lemmata give Formula (4). Lemma 3.4 gives a constraint on the types of the $(X_2, \bar{X}_2)$ and $(X_1, \bar{X}_1)$.

Lemma 3.4. *If $(X_2, \bar{X}_2)$ is of type I, then $(X_1, \bar{X}_1)$ is of type I. Moreover $\beta_2^0(X_1) = \beta_2^0(X_2)$.*

Proof. Suppose that $(X_1, \bar{X}_1)$ is of type II. Then,

$$\beta_2^0(X_1) = \alpha_2(X_1) < \frac{1}{2}\alpha_1(X_1). \tag{5}$$

$(X_1, \bar{X}_1)$ being a 1-minimum cut in $\mathscr{C}_e$

$$\frac{1}{2}\alpha_1(X_1) \leqslant \frac{1}{2}\alpha_1(X_2). \tag{6}$$

Since $(X_2, \bar{X}_2)$ is of type I, we also have

$$\beta_2^0(X_2) = \frac{1}{2}\alpha_1(X_2) \leqslant \alpha_1(X_2). \tag{7}$$

Equations (5) and (6) yield $\beta_2^0(X_1) < \frac{1}{2}\alpha_1(X_2)$. Using Equation (7) we obtain

$$\beta_2^0(X_1) < \beta_2^0(X_2).$$

This contradicts the fact that $(X_2, \bar{X}_2)$ is a 2-minimum cut. Hence, $(X_1, \bar{X}_1)$ is of type I.

Moreover, in this case, since $\beta_2^0(X_2) \leqslant \beta_2^0(X_1)$, we have $\frac{1}{2}\alpha_1(X_2) \leqslant \frac{1}{2}\alpha_1(X_1)$. Given that $(X_1, \bar{X}_1)$ is 1-minimum cut in C_e we have $\alpha_1(X_2) = \alpha_1(X_1)$, thus $\beta_2^0(X_2) = \beta_2^0(X_1)$. ■

Lemma 3.5 is a useful result that will be used in the proposition below.

Lemma 3.5. *Let $(A, \bar{A})$ be a cut in G^0, of any type, separating s and t and containing e ($(A, \bar{A}) \in C_e$). For any cut $(B, \bar{B}) \in C_e$ satisfying*

$$\beta_1^0(B) \geqslant \beta_1^0(A) \text{ and } \beta_2^0(B) \geqslant \beta_2^0(A),$$

we have

$$\forall \lambda \geqslant 0\ \beta_2^\lambda(B) \geqslant \beta_2^\lambda(A). \tag{8}$$

Proof. Four cases respective to types I and II of $(A, \bar{A})$ and $(B, \bar{B})$ may occur.

$(B, \bar{B})$ *of type I*, $(A, \bar{A})$ *of any type*: By assumption, we have
$\alpha_1(B) \geqslant \alpha_1(A)$.

$(A, \bar{A})$	$(B, \bar{B})$
I	I
I	II
II	I
II	II

Using Lemma 3.1, we have

$$\forall \lambda \geqslant 0 \ \beta_2^{\lambda}(B) = \min\left\{\alpha_2(B) + \lambda, \frac{1}{2}(\alpha_1(B) + \lambda), \alpha_1(B)\right\}$$

Since B is of type I, for $\lambda \leqslant \alpha_1(B)$, the previous equation leads to the following:

$$\begin{aligned}\beta_2^{\lambda}(B) &= \frac{1}{2}(\alpha_1(B) + \lambda)\\ &\geqslant \frac{1}{2}(\alpha_1(A) + \lambda)\\ &\geqslant \min\{\alpha_2(A), \frac{1}{2}(\alpha_1(A) + \lambda), \alpha_1(A)\}\\ &\geqslant \beta_2^{\lambda}(A)\end{aligned}$$

For $\lambda > \alpha_1(B)$, using Lemma 3.1, both $\beta_2^{\lambda}(B)$ and $\beta_2^{\lambda}(A)$ are equal to the constant term, and Equation (8) is also verified.

$(B, \bar{B})$ *of type II*: Using the previous case, we can state that

$$\forall \lambda \geqslant 0 \ \min\left\{\frac{1}{2}(\alpha_1(A) + \lambda), \alpha_1(A)\right\} \leqslant \min\left\{\frac{1}{2}(\alpha_1(B) + \lambda), \alpha_1(B)\right\} \tag{9}$$

$(A, \bar{A})$ *of type I*: Using Lemma 3.1, we have

$$\forall \lambda \geqslant 0 \ \beta_2^{\lambda}(A) = \min\left\{\frac{1}{2}(\alpha_1(A) + \lambda), \alpha_1(A)\right\}$$

Thus, it remains to show that

$$\beta_2^{\lambda}(A) \leqslant \alpha_2(B) + \lambda.$$

This property is clearly true due to the respective slopes of $\alpha_2(B) + \lambda$ (1) and $\frac{1}{2}(\alpha_1(A) + \lambda)(\frac{1}{2})$, and the fact that $\alpha_2(B) \geqslant \frac{1}{2}\alpha_1(A)$.

$(A, \bar{A})$ *of type II*: Note that for $\lambda < \alpha_1(B) - 2\alpha_2(B)$, we have

$$\beta_2^{\lambda}(B) = \alpha_2(B) + \lambda.$$

In this interval, using the general assumption of the initial 2-route flow values, we have

$$\begin{aligned}\alpha_2(B) + \lambda &\geqslant \alpha_2(A) + \lambda\\ &\geqslant \min\{\alpha_2(A), \frac{1}{2}(\alpha_1(A) + \lambda), \alpha_1(A)\}\\ &\geqslant \beta_2^{\lambda}(A)\end{aligned}$$

For $\lambda \geqslant \alpha_1(B) - 2\alpha_2(B)$, Equation (9) states Equation (8).

Thus, Equation (8) is verified and Lemma 3.5 is proved. ■

With these lemmata, the following proposition for the sensitivity analysis can be proved. Recall that $(X_2, \bar{X_2})$ is a 2-minimum cut, and $(X_1, \bar{X_1})$ is a 1-minimum cut in $\mathscr{C}_e$, separating s and t in G^0.

Proposition 3.6. $\forall \lambda \geqslant 0, \forall (Y, \bar{Y}) \in \mathscr{C}_e$, *of any type, we have*

$$\min\{\beta_2^\lambda(X_2), \beta_2^\lambda(X_1)\} \leqslant \beta_2^\lambda(Y).$$

Proof. Given the types of $(X_2, \bar{X}_2)$ and $(X_1, \bar{X}_1)$ four cases may also occur.

No.	$(X_1, \bar{X}_1)$	$(X_2, \bar{X}_2)$
(i)	I	I
(ii)	I	II
(iii)	II	I
(iv)	II	II

Case (i): By Lemma 3.4, since $(X_2, \bar{X}_2)$ is of type I, we have

$$\beta_2^0(X_2) = \beta_2^0(X_1) = \frac{1}{2}\alpha_1(X_2) = \frac{1}{2}\alpha_1(X_1). \tag{10}$$

Let $(Y, \bar{Y}) \in C_e$ be a cut. Since $(X_2, \bar{X}_2)$ is a 2-minimum cut, $(X_1, \bar{X}_1)$ is a 1-minimum cut in $\mathscr{C}_e$ and Equation (10), we have

$$\beta_2^0(Y) \geqslant \beta_2^0(X_2)$$

$$\alpha_1(Y) \geqslant \alpha_1(X_1) = \alpha_1(X_2)$$

Thus (by Lemma 3.5) $\beta_2^\lambda(Y) \geqslant \beta_2^\lambda(X_2) \forall \lambda \geqslant 0$. Hence,

$$\beta_2^\lambda(Y) \geqslant \min\{\beta_2^\lambda(X_2), \beta_2^\lambda(X_1)\} \forall \lambda \geqslant 0.$$

Case (ii): Since $(X_2, \bar{X}_2)$ is a 2-minimum cut of type II

$$\beta_2^0(X_2) = \alpha_2(X_2) < \frac{1}{2}\alpha_1(X_2).$$

Moreover, since $(X_1, \bar{X}_1)$ is a 1-minimum cut in $\mathscr{C}_e$ of type I

$$\alpha_2(X_2) \leqslant \frac{1}{2}\alpha_1(X_1) \leqslant \frac{1}{2}\alpha_1(X_2).$$

Let $(Y, \bar{Y}) \in \mathscr{C}_e$ be a cut.

$(Y, \bar{Y})$ *is of type I*: Since $(X_1, \bar{X}_1)$ is of type I, we have

$$\beta_2^0(Y) = \frac{1}{2}\alpha_1(Y) \geqslant \frac{1}{2}\alpha_1(X_1) = \beta_2^0(X_1).$$

Thus, by Lemma 3.5,

$$\beta_2^\lambda(Y) \geqslant \beta_2^\lambda(X_1) \forall \lambda \geqslant 0.$$

Hence, $\beta_2^\lambda(Y) \geqslant \min\{\beta_2^\lambda(X_2), \beta_2^\lambda(X_1)\} \forall \lambda \geqslant 0$.

$(Y, \bar{Y})$ *is of type II*: Two subcases may occur.
ii(a): $\alpha_2(Y) \geqslant \frac{1}{2}\alpha_1(X_1)$: In this case Lemma 3.5 gives

$$\beta_2^\lambda(Y) \geqslant \beta_2^\lambda(X_1) \forall \lambda \geqslant 0.$$

Thus, $\beta_2^\lambda(Y) \geqslant \min\{\beta_2^\lambda(X_2), \beta_2^\lambda(X_1)\}$.
ii(b): $\alpha_2(Y) < \frac{1}{2}\alpha_1(X_1)$. If $\alpha_1(Y) \geqslant \alpha_1(X_2)$ Lemma 3.5 also gives

$$\beta_2^\lambda(Y) \geqslant \beta_2^\lambda(X_2) \forall \lambda \geqslant 0.$$

Thus, $\beta_2^\lambda(Y) \geqslant \min\{\beta_2^\lambda(X_2), \beta_2^\lambda(X_1)\}$.
Otherwise, let us represent the variations of $\beta_2^\lambda(Y)$, $\beta_2^\lambda(X_1)$ and $\beta_2^\lambda(X_2)$ in Fig. 2.
Let μ be the *x-axis* of the intersection point of the lines of equations

$$\alpha_2(X_2) + \lambda \text{ and } \frac{1}{2}[\alpha_1(X_1) + \lambda],$$

d the line joining the points $(0, \alpha_2(Y))$ and $(\mu, \alpha_2(X_2) + \mu)$, and v the *x-axis* of the intersection point of the line d and the line of equation $\frac{1}{2}[\alpha_1(Y) + \lambda]$.

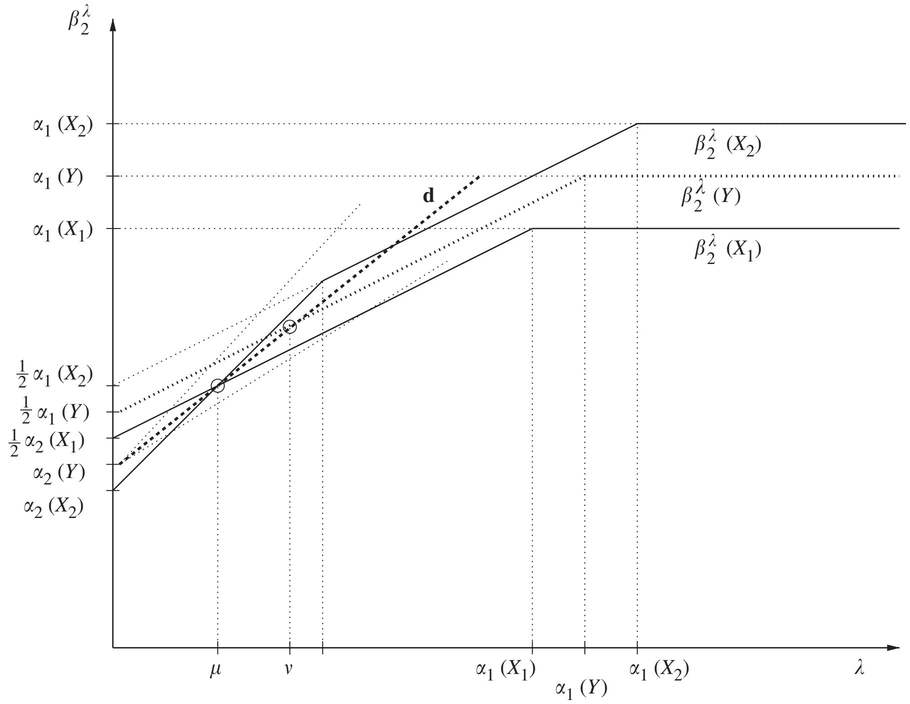

Fig. 2. Variations of $\beta_2^\lambda(y), \beta_2^\lambda(X_1)$, and $\beta_2^\lambda(X_2)$.

One can observe that, as long as the line $\alpha_2(Y)+\lambda$ (dot line above d) stays above d, we will have the target result $\min\{\beta_2^{\lambda}(X_2),\beta_2^{\lambda}(X_1)\}\leqslant\beta_2^{\lambda}(Y)$.

Thus, in order to finish the proof we have to show that the line $\alpha_2(Y)+\lambda$ is always above the line d. This is equivalent to showing that the slope of d is always lower than the slope of $\alpha_2(Y)+\lambda$.

It can be verified that, by definition, the exact value of μ is

$$\mu=\alpha_1(X_1)-2\alpha_2(X_2)\;>\;0. \tag{11}$$

Thus the slope of d is given by

$$\frac{\alpha_2(X_2)+\mu-\alpha_2(Y)}{\mu}=\frac{\alpha_1(X_1)-\alpha_2(X_2)-\alpha_2(Y)}{\alpha_1(X_1)-2\alpha_2(X_2)}\;>\;0.$$

Since the slope of $\alpha_2(Y)+\lambda$ is 1, let us suppose, by contradiction, that

$$\frac{\alpha_1(X_1)-\alpha_2(X_2)-\alpha_2(Y)}{\alpha_1(X_1)-2\alpha_2(X_2)}\;>\;1,$$

(i.e., the slope of d is greater than the slope of $\alpha_2(Y)+\lambda$).

Then, it may be deduced that

$$\alpha_2(X_2)\;>\;\alpha_2(Y).$$

This is impossible since $(Y,\bar{Y})$ and $(X_2,\bar{X}_2)$ are of type II and $(X_2,\bar{X}_2)$ is a 2-minimum cut. Thus,

$$\frac{\alpha_1(X_1)-\alpha_2(X_2)-\alpha_2(Y)}{\alpha_1(X_1)-2\alpha_2(X_2)}\leqslant 1,$$

and the result is proved.

Case (iii): Using Lemma 3.4, this case cannot occur.

Case (iv): Let $(Y,\bar{Y})\in C_e$ be a cut.

If $(Y,\bar{Y})$ *is of type I*, since $(X_1,\bar{X}_1)$ is of type II we have

$$\beta_2^0(Y)=\frac{1}{2}\alpha_1(Y)\geqslant\frac{1}{2}\alpha_1(X_1)\;>\;\alpha_2(X_1)=\beta_2^0(X_1).$$

Moreover $\beta_1^0(Y)\;>\;\beta_1^0(X_1)$.

Thus, by Lemma 3.5, we deduce $\beta_2^{\lambda}(Y)\geqslant\beta_2^{\lambda}(X_1)$. Hence,

$$\beta_2^{\lambda}(Y)\geqslant\min\{\beta_2^{\lambda}(X_2),\beta_2^{\lambda}(X_1)\}\forall\lambda\geqslant 0.$$

If $(Y,\bar{Y})$ *is of type II*, let us consider the following two subcases:

$$\triangleright\alpha_2(Y)\geqslant\alpha_2(X_1).$$

Since $\beta_2^0(Y)=\alpha_2(Y)$ (i.e., $(Y,\bar{Y})$ of type II) and $\beta_2^0(X_1)=\alpha_2(X_1)$ (i.e., $(X_1,\bar{X}_1)$ of type II) we have $\beta_2^0(Y)\geqslant\beta_2^0(X_1)$.

In addition, since $(X_1,\bar{X}_1)$ is a 1-minimum cut in $\mathscr{C}_e$, $\beta_1^0(Y)\geqslant\beta_1^0(X_1)$.

Thus, with Lemma 3.5, we deduce that $\beta_2^{\lambda}(Y) \geqslant \beta_2^{\lambda}(X_1)$; hence,

$$\beta_2^{\lambda}(Y) \geqslant \min\{\beta_2^{\lambda}(X_2), \beta_2^{\lambda}(X_1)\} \forall \lambda \geqslant 0.$$

$\triangleright \alpha_2(Y) < \alpha_2(X_1)$

For $\alpha_1(Y) \geqslant \alpha_1(X_2)$, $\beta_1^0(Y) \geqslant \beta_1^0(X_2)$. Since we also have

$$\beta_2^0(Y) \geqslant \beta_2^0(X_2) \ (i.e.(X_2, \bar{X}_2) \text{ a 2-minimum cut}),$$

it follows with Lemma 3.5 that $\beta_2^{\lambda}(Y) \geqslant \beta_2^{\lambda}(X_2)$; hence,

$$\beta_2^{\lambda}(Y) \geqslant \min\{\beta_2^{\lambda}(X_2), \beta_2^{\lambda}(X_1)\} \forall \lambda \geqslant 0.$$

For $\alpha_1(Y) < \alpha_1(X_2)$, we obtain a similar graphic as for the Case ii(b).

Thus, using the same definitions and arguments, the sketch of the proof is exactly the same as for Case ii(b). ■

Therefore, the 2-minimum cuts $(X_2, \bar{X}_2)$ and $(X_1, \bar{X}_1)$, computed for $\lambda = 0$, are for each λ always lower or equal to the 2-capacity of the other cuts containing e, whatever their types are. Thus, in the sensitivity analysis, especially for the set $\mathscr{C}_e$, only these cuts are of interest. The theorem below gives the formulation of $f_{s,t}^2(\lambda)$.

Theorem 3.7. *Let $G = (N, E, c)$ be a network, and $e \in E$ an edge with a parameterized capacity $\lambda \geqslant 0$. Let (s, t) be a source–sink pair of vertices of G.*

The maximum flow value $f_{s,t}^2(\lambda)$ verifies

$$f_{s,t}^2(\lambda) = \min\{\beta_2^{\lambda}(X_2), \beta_2^{\lambda}(X_1), f_{s,t}^2(\infty)\}.$$

Proof. By Proposition 3.6, we have

$$\min_{C \in C_e} \beta_2^{\lambda}(C) = \min\{\beta_2^{\lambda}(X_2), \beta_2^{\lambda}(X_1)\}.$$

By the Equation (4), following Lemma 3.3, we also have

$$f_{s,t}^2(\lambda) = \min\{\min_{C \in C_e} \beta_2^{\lambda}(C), f_{s,t}^2(\infty)\} \forall \lambda \geqslant 0.$$

Thus, we have

$$f_{s,t}^2(\lambda) = \min\{\beta_2^{\lambda}(X_2), \beta_2^{\lambda}(X_1), f_{s,t}^2(\infty)\}. \quad ■$$

For all source–sink pair (s, t) of G, computing efficiently the value $f_{s,t}^2(\lambda)$ is possible as follows. First recall that G^0 is the network in which $c(e) = 0$ and G^M the network in which $c(e) = M$. The value M is computed in Corollary 3.2 and $f_{s,t}^2(\lambda)$ stagnates for $\lambda > M$. Thus, computing $f_{s,t}^2(\lambda)$ may be done.

- by using the Gomory–Hu trees, T_0 and T_M, of G^0 and G^M,
- by deriving the corresponding 2-cut-trees T'_0 and T'_M,
- and finally performing $O(n^2)$ operations in which for each pair (s, t) the theorem above is used.

These steps are summarized in the algorithm of Fig. 3.

Data

- $\lambda \geq 0$: a parameter
- $G^{\lambda} = [N, E, c]$: a network in which $c(e) = \lambda$ where $e \in E$
- M : An arbitrary big value

Init

1. Compute a Gomory-Hu Tree T_0 (resp. T_M) of G^0 (resp. G^M)

2. Compute from T_0 (resp. T_M) a 2-cut-tree $T_0^{'}$ (resp. $T_M^{'}$) (Theorem 2.6)

Loop

For all source-sink pair (s, t)

Compute with $T_0^{'}$ *a* $2-$ *minimum* $s - t$ *cut* $(X_2, \overline{X_2})$ (Lemma 2.5)
Compute with $T_M^{'}$ *a* $2-$ *minimum* $s - t$ *cut* $(X_\infty, \overline{X_\infty})$ (Lemma 2.5)
$f_{s,t}^{2}(\infty) = \beta_2^M(X_\infty)$
Compute with T_0 *a* $1-$ *minimum* $s - t$ *cut* $(X_1, \overline{X}_1)$ *in* X_e
Apply Theorem 3.7 *to find* $f_{s,t}^{2}(\lambda)$

Fig. 3. Sensitivity analysis algorithm.

Finding Gomory–Hu trees (step 1) and deriving 2-cut-trees (step 2) need a polynomial number of operations. In the same vein, the loop also induces a polynomial number of operations. The whole method proposed here is therefore strongly polynomial.

4. Analyzing the effect of several parameterized capacities

In this section, we study the case where $k \geqslant 2$ edges have parameterized capacities. Let us suppose that in G, the edges $e_1, e_2, \ldots, e_k$ have non-negative parameterized capacities $\lambda_1, \lambda_2, \ldots, \lambda_k$ (i.e., $c(e_1) = \lambda_1$, $c(e_2) = \lambda_2, \ldots, c(e_k) = \lambda_k$). Let (s, t) be a source–sink pair of nodes. As previously, the goal of the sensitivity analysis for this case is to find an efficient technique to compute the maximum 2-route flow values between s and t. We denote by $f_{s,t}^2(\lambda_1, \lambda_2, \ldots, \lambda_k)$ this flow value.

The following results generalized the method proposed in the previous section.

Corollary 4.1. *Let* $G = (V, E)$ *be a network and* $e_1, e_2, \ldots, e_k$ *be k different edges with parameterized capacities. All pairs maximum flow values* $f_{s,t}^2(\lambda_1, \lambda_2, \ldots, \lambda_k)$ *for all parameters can be computed using* 2^k *1-cut-tree if the capacity of the edges vary independently.*

Proof. The result is obtained with a recurrence as in Berthomé et al. (2003). The result is true for $k = 1$ by Theorem 3.7.

Let us consider that it is still true until $k - 1$ and let us prove that it is also true for k.

If we consider that one of the k parameters, say λ_1, is fixed, with the recurrence hypothesis we know that 2^{k-1} 1-cut-tree are necessary to compute all flow values for any assignment of the other

$k-1$ parameters. Then it remains to compute each maximum flow with respect to the last dimension λ_1. Since, by Theorem 3.7, for each computation two 1-cut-tree are necessary we obtain finally 2^k 1-cut-tree. ■

5. Conclusion

In this work, we have shown how to compute efficiently all maximum 2-route flow values, in the case of an indirect network with the presence of parameterized edge capacities. This theoretical and algorithmic study, to the best of our knowledge, is the first in the q-route flow setting. The expressions proposed above may be useful in the context of the 2-realizability problem (Kabadi et al., 2005) in which an application may be found.

Acknowledgments

The work of the first author is funded by Research Productivity Grant 2008–2011 by CNPq, Brazil.

References

Aneja, Y.P., Chandrasekaran, R., Kabadi, S.N., Nair, K.P.K., 2007. Flows over edge-disjoint mixed multipaths and applications. *Discrete Applied Mathematics* 155, 1979–2000.

Arggawal, C.C., Orlin, J.B., 2002. On multiroute flows in networks. *Networks* 39, 43–52.

Barth, D., Berthomé, P., Diallo, M., Ferreira, A., 2006. Revisiting parametric multiterminal problems: maximum flows, minimum cuts and cut-tree computations. *Discrete Optimization* 3, 3, 195–205.

Berthomé, P., Diallo, M., Ferreira, A., 2003. Generalized parametric multi-terminal flows problem. In Bodlaender, H.L. (eds) *Graph-Theoretic Concepts in Computer Science, Lecture Notes in Computer Science*, Vol. 2880, Springer, Berlin, pp. 71–80.

Du, D., 2003. Multiroute flow problem. Ph.D. thesis, University of Texas at Dallas, Dallas, TX, USA.

Du, D., Chandrasekaran, R., 2006. Multiroute maximum flow revisited. *Networks* 47, 2, 81–92.

Du, D., Kabadi, S.N., 2005. An improved algorithm for decomposing arc-flows into multipath-flows. *Operations Research Letters* 34, 1, 53–57.

Elmaghraby, S.E., 1964. Sensitivity analysis of multi-terminal network flows. *Journal of the Operations Research Society of America* 12, 680–688.

Ford, L.R., Fulkerson, D.R., 1973. *Flows in Network*. Princeton University Press, Princeton, NJ.

Gomory, R.E., Hu, T.C., 1961. Multi-terminal network flows. *SIAM Journal of Computing* 9, 551–570.

Kabadi, S.N., Chandrasekaran, R., Nair, K.P.K., 2005. Multiroute flows: cut-trees and realizability. *Discrete Optimization* 2, 3, 229–240.

Kishimoto, W., 1996. A method for obtaining maximum multi-route flows in an network. *Networks* 27, 229–291.

Kishimoto, W., 1997. Reliable flow with failures in a network. *IEEE Transactions on Reliability* 46, 308–315.