Maximum Delay Computation under Traffic Matrix Uncertainty and its Application to Interdomain Path Selection

Isabel Amigo, Sandrine Vaton, Thierry Chonavel, and Federico Larroca

Abstract One of the most important problems when deploying interdomain path selection with quality of service requirements is being able to rely the computations on metrics that hold for a long period of time. Our proposal for achieving such assurance is to compute bounds on the metrics, taking into account the uncertainty on the traffic demands. In particular, we will explore the computation of the maximum end-to-end delay of traversing a domain considering that the traffic is unknown but bounded. Since this provides a robust quality of service value for traversing the Autonomous System (AS), without revealing confidential information, we claim that the bound can be safely conceived as a metric to be announced by each AS to the entities performing the path selection, in the process of interdomain path selection. We show how the maximum delay value is obtained for an interdomain bandwidth demand and we propose an exact method for solving the optimization problem. Simulations with real data are also presented.

1 Introduction

New Internet market proposals are emerging, mainly due to the new technological offers and the positioning towards them of all the involved actors [?]. Value-added services with real time requirements, such as videoconferencing and telepresence, are showing up as the new stars of the Internet for Service Providers (SPs) and Customers. The Network Providers interests are moving towards obtaining new revenues and business opportunities out of these new services, that rely completely on their deployed network infrastructure. Network Providers must be able to assure Quality of Service (QoS), so as to fulfil the Customers expectations and to be able to trade among SPs, for instance by means of Service Level Agreements (SLAs), which means, in turn, that SPs can offer better services to their Customers.

This arising scenario becomes more complex when the services provided traverse several domains, or Autonomous Systems (ASs), in its way from the SP to the Customer. In this case, QoS must be provided all throughout the path, involving different ASs, which raises several technical, economical and political issues. Concerning the technical aspect, achieving scalability, preserving confidentiality and providing interoperability is of paramount importance in any technical solution [?].

In the framework of an alliance of ASs, carriers work together in order to achieve a common interest. In this scenario QoS values related to each domain are exchanged, and Traffic Engineering decisions are taken according to them. Different mechanisms have been proposed for the selection and establishment of interdomain QoS constrained tunnels, that mainly rely on RSVP-TE [?] and the PCE architecture [?] (e.g. [?, ?, ?]). These mechanisms are based on metrics announced by each AS but they do not specify how to compute such metrics. The complexity resides mainly in the fact that the announced metrics have to hold for some period of time, ideally as long as the service is provided. Hence, ASs must be able to provide QoS values that are guaranteed to hold for a certain period of time.

We shall put our focus on point-to-point services with QoS requirements. In this case the service may be abstracted to a QoS guaranteed tunnel (for instance an MPLS tunnel [?]). The path traversed by the tunnel must fulfil the QoS parameters required by the service.

In particular, our attention will be focused on those services for which available bandwidth and end-to-end delay are critical parameters. The end-to-end delay is compounded of the sum of the delays introduced by each transit AS and the terminal ones, from source to destination. As illustrated in Fig. ??, where we show a situation with two terminal ASs and one transit AS, the delay in each of the ASs depends on the traffic already present in the AS (t_* flows in Fig. ??), the topology, the routing configuration, and the traffic coming from the new tunnel (flow u in Fig. ??).

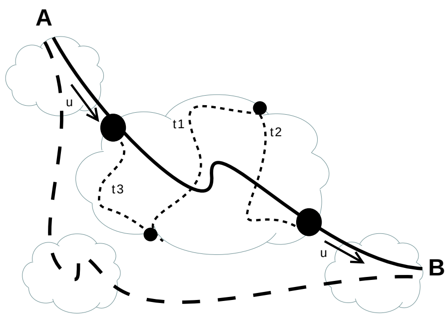


Fig. 1 Scenario

But, why do not simply advertise an instantaneous value of the metric? The main fact that makes an instantaneous value of the metric a non appropriate one is the existence of uncertainty, which implies that the value can change in the immediate future. We could, however, follow a dynamic approach, in which network state is continuously monitored and metric value is updated. These reactive approaches make it possible to tightly follow the variations of the traffic but they require a monitoring infrastructure to be present and some sophisticated algorithms to process the measurement data. Moreover, reactive approaches are able to detect variations in the traffic demand such as abrupt changes but they are not able to forecast them [**?**]. On the contrary, proactive mechanisms provide pessimistic values of QoS metrics but they are able to provide metrics values which are likely to hold for a given period of time since in that case uncertainty is taken proactively into account. In this work we will use the robust approach, in which a bound for the metric is provided.

In this context, uncertainty can be classified into two types: network state uncertainty and traffic uncertainty. Uncertainty in network state refers to the situation where the topology changes or is partially known. This may be due to information arriving out of date or not synchronized to the entity performing the computation, or simply to link failures. In the literature some approaches have been proposed for performing QoS routing under this kind of uncertainty [**?**, **?**, **?**]. However, in the present paper we will assume that the topology does not change, and considering this uncertainty is left for future work.

On the other hand, we will consider uncertainty in the traffic. This refers to the fact that the flows traversing the domain are not perfectly known. This can be due to the fact that changes occur rather frequently. The reason of these changes may be several, for instance, external routing modification, the presence of unexpected events such as network equipment failures outside the domain, large-volume network attacks or flash crowd occurrences [**?**].

In summary, we shall focus on the computation of a bound for the end-to-end delay of traversing an AS, from a given Origin to a Destination node, as a function of the AS parameters we mentioned before: the routing configuration, the traffic demands and the traffic injected through the new tunnel. We will consider the situation where traffic variation is the principal cause of delay variation and we will assume that the topology and the routing configuration are fixed. However, we will consider that traffic is non-static, and that it is contained in a so-called uncertainty set [**?**]. The question of how to choose this set is discussed later in the paper.

2 Problem Statement

In this section we formally present the problem of finding the maximum end-to-end delay experienced by a bounded amount of traffic traversing an AS through a particular path. As mentioned before, we will consider that traffic varies within an uncertainty set. First, let us introduce the notations that are going to be used throughout the paper and state some assumptions.

2.1 Assumptions and Notations

The network is compounded of n nodes and of a set L of links, $L = \{l_1 \ldots l_{|L|}\}$, where the notation $|\cdot|$ refers to the cardinality of the set. Traffic demands will be represented by the so-called traffic matrix $TM = \{tm_{i,j}\}$, where $tm_{i,j}$ is the amount of traffic from node i to node j. We shall use as well the term Origin Destination (OD) flows to refer to them. We reorder every traffic demand and rewrite the OD flows $(tm_{i,j})$ in vector form as t, $t = \{t_k\}$, $k = 1 \ldots n(n-1)$. The amount of traffic coming from the interdomain injected into the new tunnel will be u.

The link load Y is a vector containing in the i-th entry the load on link i without considering u. With these definitions we can see that $Y = R.t$ where R, a $|L| \times m$ matrix $(m = n(n-1))$, is the routing matrix, which means that $\{R_{i,j}\} = 1$ if flow j traverses link i, and 0 otherwise.

The flow that carries u will traverse the AS from an origin to a destination node following a certain path. We will call this path P. We will equally refer to the set of links that belong to that path as P, in this case P is a subset of L.

The mean link delay is approximated by the M/M/1 model, that is to say $D_l = \frac{K}{c_l - y_l}$, where c_l is the capacity of the link l and K the mean packet size. We then approximate the mean delay of a path by the sum of the delays of the links it traverses:

$$Delay_P = \sum_{l \in P} \frac{K}{c_l - y_l}. \tag{1}$$

The propagation delay may be ignored in our formulation since it does not change with the load and may be added as a constant later on. Moreover, the M/M/1 model is used for illustrative purposes only. In fact, any convex function may be used instead. See [**?**] on how to obtain a good approximation of the delay function based on measurements. We will as well ignore the constant K in the following formulations, for the sake of notations simplification.

2.2 Modelling Traffic Uncertainty

As mentioned above, we will not make any assumptions on the traffic matrix except that it always belongs to a certain uncertainty set. In particular we will follow the approach presented in [**?**] and define the uncertainty set as a polytope formed by the result of the intersection of several half-spaces. Consequently, all constraints can be written as $A \times t \leq b$, where A is a certain matrix which can be defined after different models, and b is a given bound. We now present four examples of polytope definition.

The Hose Model

This particular case of the general polytope definition was presented in [**?**] in the context of VPN services specification. It establishes that the input and output total traffic on each node is bounded. That is to say: $\sum_i tm_{i,j} \leq b_j^+$ and $\sum_i tm_{j,i} \leq b_j^- \;\forall\; i \in N,\, j \in \{N \setminus i\}$, where b_j^-, b_j^+ are given bounds on the total ingress and egress traffic and N is the set of network nodes.

Links Capacity Model

This model results of the application of bounds on the total traffic traversing the different links of the network, $y_i \leq b_i$. These constraints can also be written as $R^h \times t \leq b$, where $b = \{b_i\}$ are historical maximums taken for instance form measurements, and R^h is the routing matrix at the moment when the measurements were taken. This approach is used for example in [**?**] where a polyhedral definition of the traffic matrix is preferred to its estimation because of non stationarity artifacts and estimation errors.

Known Statistical Values

If mean, variance and covariance values of link loads are known, we can compute the variance ellipsoid as $\{w = \varrho + \alpha \mid \alpha^T \Omega \alpha \leq 1\}$ where ϱ is the expected value of the link loads, and Ω its covariance matrix. Therefore, the variables w describe an ellipsoid. Several half-planes tangent to the ellipsoid can be defined in order to obtain linear constraints. Figure **??** illustrates this example. The polytope can then be written as $A \times R \times t \leq b$, were R is the routing matrix and A and b define the polytope in which the ellipsoid is inscribed.

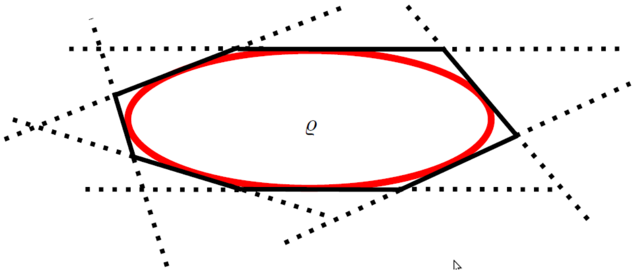


Fig. 2 Example for defining a polytope after known statistical values.

Prediction Based Model

This model consists of defining bounds on the value of traffic demands which are based on traffic prediction. The prediction of future demands is based on past observations. For example artificial intelligence methods such as neural networks or time series analysis can be used in order to forecast the future values of the traffic demand; see for example [?] for prediction based on a seasonal ARIMA model.

2.3 Mathematical Formulation

The problem consists on, given a path, computing the maximum end-to-end delay of that path, allowing the traffic matrix t to vary within a polytope. That is to say that we will work with a maximization problem with linear constraints. Let us introduce the m-dimensional column vector w_l, $l \in P$, as $w_l = \{w_{l,i}\} = R_{l,i}/c_l$.

The optimization problem is described by Problem **??**, where A and b define the polytope.

Problem 1.

$$\begin{array}{ll} \max\limits_{t} & \sum\limits_{l \in P} 1/c_l \dfrac{1}{1 - w_l^T t - u/c_l} \\ \text{s.t.} & A \times t - b \leq 0. \end{array}$$

Please note that if some additional linear constraints must be taken into account they can be integrated in the definition of the polytope $A \times t \leq b$. Example of such constraints can be $w_l^T t + u/c_l < 1$, for $l \in P$, which simply states that there should be enough link capacity in order to accommodate all the traffic, including the new tunnel.

The objective function in the maximization problem defined by Problem **??** is not a concave function, consequently, the problem is not a convex one. On the contrary, the problem is the maximization of a convex function over a polytope. This is a very difficult problem, all the more so since the objective function is not strictly convex.

Intuitively we can see that the function is not strictly convex due to the difference between the number of links and the number of OD flows. Indeed, while the number of links grows linearly with the number of nodes in the network, the number of OD flows squares with the number of nodes in the network. This means that for different values of the vector t the objective function of Problem **??** can have the same value, while its gradient remains always non-negative.

More formally, we state the following proposition.

Proposition 1. *The function $f(t)$, objective function of Problem* **??***, is a convex function over the set $S = \{t \in \mathbb{R}^m | A \times t \leq b\}$, but not a strictly convex one.*

Proof. We explore if the following inequality holds [**?**]

$$f(t_1) \geq f(t_2) + \nabla f(t_2)^T (t_1 - t_2),\ t_1, t_2 \in S. \tag{2}$$

Applying the definition of f to Eq. (**??**) we obtain the following inequality for $t_1,\ t_2 \in S$:

$$\sum_{l \in P} \frac{1/c_l}{1 - w_l^T t_1 - u/c_l} \geq \sum_{l \in P} \frac{1/c_l}{1 - w_l^T t_2 - u/c_l} + \sum_{l \in P} \frac{1/c_l \times w_l^T (t_1 - t_2)}{(1 - w_l^T t_2 - u/c_l)^2}. \tag{3}$$

Let us now define $g_l(t)$, an auxiliary function in order to simplify the notations, as

$$g_l(t) = 1 - w_l^T t - u/c_l,\ t \in S. \tag{4}$$

Substituting the latter definition in Eq. (**??**) and performing some regular math operations we obtain the following inequality

$$\sum_{l \in P} \frac{(g_l(t_2) - g_l(t_1))^2}{g_l(t_1) g_l(t_2)^2} \geq 0,\ t_1, t_2 \in S. \tag{5}$$

Each term on Inequality (**??**) is either zero or greater than zero for all $t_1,\ t_2 \in S$. Therefore, the function f is convex over S. It remains to show if the function is strictly convex or not. Which is equivalent to showing if there exist t_1 and $t_2 \in S$ such that $< w_l, t_2 - t_1 >$ is equal to zero for all $l \in P$, that is to say, having all vectors w_l, $l \in P$ orthogonal to the vector $(t_2 - t_1)$, or not. Since the vectors w_l do not form a basis of $\mathbb{R}^m$ it is possible to find t_1 and $t_2 \in S$ such that their difference is orthogonal to all vectors w_l, $l \in P$. □

Proposition **??** showed that f is a convex function, but not a strictly convex one. However, in the following section we reformulate the problem and show a way to find its solution.

3 Finding the Solution

We now state the problem in a different way which will allow us to find its solution. We aim at formulating the problem in such a way that the objective function is strictly convex and the dimension of the problem is reduced. For doing so we shall decompose the vector t over a particular basis of $\mathbb{R}^m$.

The procedure consists in decomposing the vector t over the vectors w_l, $l \in P$, and their orthogonal complement. We define the matrix W_1 as an m by $|P|$ matrix, whose columns are the vectors w_l, with $l \in P$, and W_2, an m by $m - |P|$ matrix such that it verifies

$$W_1^T \times W_2 = 0. \tag{6}$$

Provided that the columns of W_1 are linearly independent, it can be proven that the columns of the matrix W defined after W_1 and W_2 as

$$W = [W_1 W_2] = [w_1, \ldots, w_l, \ldots, w_{|P|}, \ldots w_m] \tag{7}$$

represent a basis of $\mathbb{R}^m$.

We shall decompose the vector t over the defined basis using the auxiliary variables $x \in \mathbb{R}^{|P|}$ and $h \in \mathbb{R}^{m-|P|}$ as

$$t = W_1 x + W_2 h. \tag{8}$$

By multiplying both sides of Eq. (**??**) by w_l^T, and using Eq. (**??**) we obtain

$$w_l^T t = w_l^T W_1 x = v_l^T x, \tag{9}$$

where we have set $v_l^T = w_l^T W_1$, for all $l \in P$. Note that both v_l and x are column vectors of dimension $|P|$.

Equation (**??**) will directly lead us to rewriting the objective function of Problem **??** as a function of x. We shall now redefine the polytope by writing it in the basis W which leads to defining a new matrix denoted D and computed as $A \times W$. The polytope over the new basis can be compactly written as $D[x^T \; h^T]^T \leq b$.

All in all, Problem **??** can be rewritten in the form of Problem **??**. Please note that the objective function depends only on the variable x.

Problem 2.

$$\begin{array}{ll} \max\limits_{x} & \sum\limits_{l \in P} 1/c_l \dfrac{1}{1 - v_l^T x - u/c_l} \\ \text{s.t.} & D \begin{pmatrix} x \\ h \end{pmatrix} \leq b \end{array}$$

Let us call the objective function of Problem **??** as $J(x)$ and the new polytope as V (i.e. $V = \left\{[x^T \ h^T]^T \in \mathbb{R}^m : D[x^T \ h^T]^T \leq b\right\}$). Let us as well define the polytope V_x as

$$V_x = \left\{x \in \mathbb{R}^{|P|} \mid \exists \ h \in \mathbb{R}^{m-|P|} : D[x^T \ h^T]^T \leq b\right\}. \tag{10}$$

Let $\mathcal{W}_1 = span\{w_1 \ldots w_{|P|}\}$, where span refers to the set of all linear combinations of vectors $w_1 \ldots w_{|P|}$. Clearly V_x is the projection of V onto $\mathcal{W}_1$.

Since V is a convex polytope by definition, it is easy to check that V_x is also a convex polytope. More precisely, V_x is the convex hull of the projection of the extreme points of V onto $\mathcal{W}_1$ [**?**].

Then, since $J(x)$ does not depend on h, Problem **??** can be represented in the space $\mathcal{W}_1$ as follows:

Problem 3.

$$\begin{array}{ll} \max\limits_{x} & J(x) \\ \text{s.t.} & x \in V_x. \end{array}$$

The following statement summarizes our development of the problem.

Proposition 2. *The optimization problem defined by Problem* **??** *is equivalent to the one defined by Problem* **??***.*

We now show that $J(x)$ is a strictly convex function over V_x, which will in turn allow us to prove that the solution of Problem **??** is attained at an extreme point of the polytope V_x.

Proposition 3. *The function $J(x)$, objective function of Problem* **??***, is a strictly convex function over the set V_x defined as in* (**??**)*.*

Proof. We define $\lambda_l(x)$ as

$$\lambda_l(x) = (1 - v_l^T x - u/c_l)^{-2}, \ \forall l \in P \tag{11}$$

and the matrix Λ as

$$\Lambda(x) = diag(\lambda_1, \ldots, \lambda_{|P|}). \tag{12}$$

For all $x \in V_x$ and $l \in \{1 \ldots |P|\}$, $\lambda_l(x) > 0$. Thus, $\Lambda(x)$ is a positive-definite matrix[1].

[1] A $n \times n$ real symmetric matrix M is positive-definite if $z^T M z > 0$ for all non-zero vectors z, $z \in \mathbb{R}^n$.

In addition, we can check that $[v_1 \dots v_{|P|}] = W_1^T W_1$ is also a positive-definite matrix. Thus, the Hessian of $J(x)$, which is

$$\nabla^2 J(x) = (W_1^T W_1)\Lambda(x)(W_1^T W_1) \tag{13}$$

is as well a positive-definite matrix. □

We are now able to show that the solution to Problem **??** is attained at an extreme point of V_x.

Theorem 1. *The solution of Problem* **??** *is attained at an extreme point of the polytope* V_x*, defined by the set* (**??**).

Proof. We prove by contradiction that the maximum of $J(x)$ over V_x must be reached at an extreme point of V_x. Since, by Proposition **??**, J is a strictly convex function, inequality (**??**) holds [**?**].

$$J(\Phi) > J(\theta) + \nabla J(\theta)^T(\Phi - \theta), \forall\ \theta, \Phi\ \in V_x. \tag{14}$$

Now, let $\bar{\theta} \in V_x$ be an optimal point of Problem **??**. Therefore, $\bar{\theta}$ is a strict maximum, since J is strictly convex, and, for all $\Phi \in V_x \setminus \{\bar{\theta}\}$, we must have:

$$J(\Phi) - J(\bar{\theta}) < 0. \tag{15}$$

Together with inequality (**??**), we get

$$\nabla J(\bar{\theta})^T(\Phi - \bar{\theta}) < 0,\ \forall \Phi \in V_x \setminus \{\bar{\theta}\}. \tag{16}$$

By contradiction we suppose that $\bar{\theta}$ is not an extreme point of V_x. Then there exists $\mu \in \mathbb{R}^{|P|}$ such that $||\mu|| > 0$ and $\bar{\theta} + \mu$, $\bar{\theta} - \mu \in V_x$. By letting $\Phi = \bar{\theta} - \mu$ and $\Phi = \bar{\theta} - \mu$ at a time, we would get:

$$\nabla J(\bar{\theta})^T \mu < 0 \text{ and } -\nabla J(\bar{\theta})^T \mu < 0, \tag{17}$$

which is not possible. □

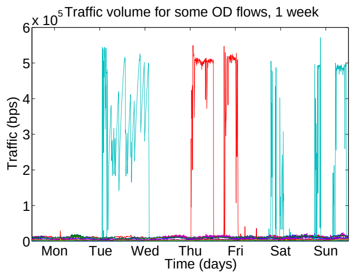
Problem **??** allows us to work with a strictly convex function, and to reduce the dimension of the feasible region, in some cases, considerably. According to Preposition **??** along with Theorem **??**, finding the extreme points of the polytope V_x renders the solution of Problem **??**. Therefore, we need to be able to perform the projection of a polytope, and afterwards enumerate its extreme points. Methods for doing so are available (see for instance [**?**]), although these can be computationally expensive tasks. In the following section we explore this solution by performing simulations in real topologies.

4 Simulations

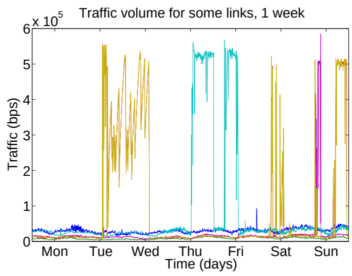
To evaluate the proposed method we present some simulation studies. The simulations are carried out using two different research networks. Namely, the Abilene network, whose topology, historical traffic demands and routing matrix are available from [?], and the GÉANT network [?]. All results are computed on a regular computer (Intel Pentium Dual-core 1.86GHz, 2GB of RAM). For computing the polytope projection and enumerating its extreme points we use the MPT library [?] and the ET library [?], distributed along with the former.

4.1 The Abilene Network

The Abilene network consists of 30 internal links and 12 routers, all exchanging traffic among them. Figure **??** shows a traffic trace of Abilene's network. In this example we can see how the traffic matrix is prone to sudden traffic variations. Figure **??** shows the traffic for some OD flows corresponding to 2016 consecutive measurements, while Fig. **??** shows the link load.



(a) Traffic volume per OD flow.



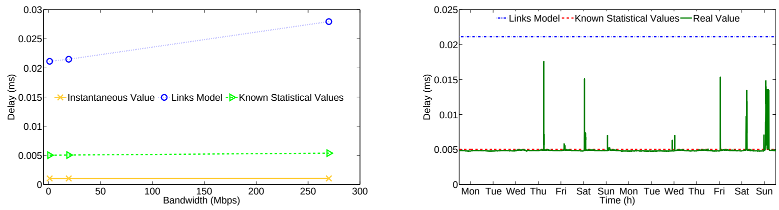
(b) Link load.

Fig. 3 Example of traffic variation in the Abilene network, one week of traffic.

For illustrative purposes we compute results for three different types of services. Namely, a VoIP service with 1 Mbps of bandwidth, a broadcast quality HDTV service with 19.4 Mbps and a VPN service with a demand of 270 Mbps. We compute the maximum delay suffered by a flow traversing the AS through a particular path and carrying each one of these services at a time. The path is chosen arbitrarily, from one origin to one destination node. Please note that this choice and its impact on the delay are out of the scope of the present paper.

In the first place, we define the polytope using the Links Load model. That is to say, the polytope is defined by imposing bounds on each link load, which are based on the maximum values obtained historically.

The values obtained for the defined path and the three services are shown in Fig.**??** (dotted line) along with the current delay value. The current delay value corresponds to a value obtained instantaneously. For this particular case the maximum delay value is approximately 3 times more than the current one which illustrates the weakness of the current value as a metric on which rely. We will come back to this kind of comparisons later on this section.



(a) Instantaneous delay value and Maximum delay value for two different polytopes and three bandwidth demands.

(b) Maximum value for two different polytopes and real values during two weeks for one bandwidth demand.

Fig. 4 Simulations in the Abilene network.

In the second simulation, we define the polytope based on the Known Statistical Values model, introduced in Sect. **??**. We compute the variance ellipsoid using a historical traffic trace (the same trace used for the first simulation) and we approximate the ellipsoid by a polytope, by intersecting several half spaces tangent to it. The maximum delay of traversing the AS is computed for the same path used in the previous simulation.

The results are shown on Fig. **??** (dashed line) for a flow traversing the same path as in the previous simulation and carrying the three defined services, one at a time. We can see that in this case the bound obtained is smaller than the one obtained in the first place and closer to the instantaneous value.

We now compare the two bounds with the real delay suffered by the path during the two weeks after the computation of the polytopes, in all the cases assuming an interdomain bandwidth demand of 1 Mbps. The results are shown in Fig. **??** which illustrates the behaviour of the bounds with respect to the real values. We can see that there is a trade-off between assuring a delay value for most of the time, by using a big polytope, or having a tighter bound most of the time, but having delays that outstrip the bound. Nevertheless, the polytope could be reduced in a safe way if we had additional information, for example by using as well the hose model which imposes bounds to the traffic coming from other clients, which may be limited by a contract and traffic shaping.

The time consumed to perform the computations varied between 48 minutes and 36 hours, which for a moderately sized network is rather high. In fact, even if in several topologies we were able to find the exact solution through these means, it is still an open question whether there exists an algorithm for enumerating all extreme points of a polytope of an arbitrary dimension in running polynomial time [?]. We will, on the next subsection, empirically explore the time consumed by the method in a larger network.

4.2 The GÉANT Network

In order to test the proposed solution on a larger topology, we use the GÉANT network. This network is compounded of 23 nodes and 74 links. Thus, we can define up to 506 independent OD flows. As we have already mentioned the computation complexity of the proposed solution is likely to grow with the dimension of the network (i.e. the number of links in the path and the number of OD flows in the network). The simulations with this network aid as to assessing the performance of the method when the number of OD flows grows. We perform the simulations considering several subsets of OD flows, containing each of them 170, 200, 230 and 260 OD flows. The polytope is defined using the Links Load model and historical data.

Figure **??** shows the time consumed by each phase of the procedure, that is to say obtaining the polytope in the new basis, projecting the polytope and finding its extreme points. We can see that in all the cases, when we increase the number of OD flows considered, the task that consumes most of the time is the projection of the polytope.

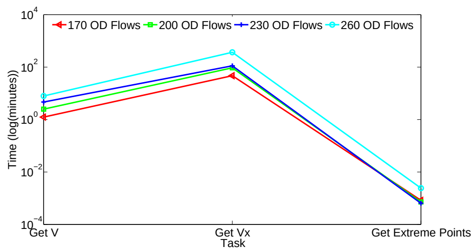


Fig. 5 Computation time as a function of the number of OD flows considered on the GÉANT Network.

The procedure has shown rather high computational times, though it was still feasible in all the tests. It is because of this that we think of this method as of great aid when developing approximated, but less time consuming, methods, since it provides the ground truth, thus a validation tool for such methods.

5 Conclusion and Future Work

In this work we have addressed the problem of the existence of uncertainties on the traffic demands in the context of interdomain QoS provisioning. The uncertainty was modeled as a polytope and different examples for building it were mentioned. We have focused our attention on the computation of a robust value of the end-to-end delay of traversing an AS under traffic uncertainty, which means obtaining a value that does not change when traffic demands do so, assuming the demands remain inside the uncertainty set. This bound was conceived as a metric to be used in the interdomain path selection process, since it provides a value that the AS can guarantee for a certain period of time, while it can be advertised without reveling confidential information. The problem was mathematically formulated and an exact solution, based on the projection of the polytope onto a subspace of smaller dimension, was proposed. Simulations with real data were performed and shown.

The theoretical study suggested that the computational times could be rather high, simulations with large network confirmed this. In order to find a remedy to this situation, we are currently studying alternative solutions based on heuristics and numerical approximation methods. The exact method proposed in this paper will be extremely useful as a tool of validation of the approximated solutions. In addition, as future work, we shall address the case of having uncertainty on the AS topology in addition to traffic uncertainty. For instance, taking into account the case of link or node failures, and being able to provide even in those cases a tight end-to-end delay bound.

Acknowledgements The research leading to these results has received funding from the Europeans Community Seventh Framework Programme (FP7/2007-2013) under grant agreement n°248567 for the ETICS Project. Further information is available at `www.ict-etics.eu`. This work was as well partially funded by the Uruguayan Agency for Research and Innovation (ANII) under grant PR-POS-2008-003.