Robust routing mechanisms for intradomain traffic engineering in dynamic networks

Pedro Casas Hernandez, Federico Larroca, Sandrine Vaton

Robust Routing Mechanisms for Intradomain Traffic Engineering in Dynamic Networks

Pedro Casas*†,Federico Larroca‡ and Sandrine Vaton†
*Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay - Email: pcasas@fing.edu.uy
†Télécom Bretagne, Brest, France - Email: firstname.lastname@telecom-bretagne.eu
‡Télécom ParisTech, Paris, France - Email: federico.larroca@telecom-paristech.fr

***Abstract*—Internet traffic is highly dynamic and difficult to predict in current network scenarios. This makes of Traffic Engineering (TE) a very challenging task for network management and resources optimization. We study the problem of Intradomain Routing Optimization under this traffic uncertainty. Recent works have proposed robust optimization techniques to tackle the problem, conceiving the Robust Routing (RR) approach. RR copes with traffic uncertainty in an off-line preemptive fashion, computing a single static routing configuration that is optimized for traffic variations within some predefined uncertainty set. Despite achieving routing reliability with relatively low performance loss, RR presents various drawbacks and conception problems as it is currently proposed. This paper brings insight into the different Robust Routing shortcomings, introducing new mechanisms that improve previous proposals and alleviate these problems. Among others, we propose and evaluate new optimization objectives to attain better global performance from an end-to-end quality of service perspective.**

***Index Terms*—Traffic Uncertainty, Proactive Traffic Management, Robust Optimization, Stable and Reactive Robust Routing, End-to-End Routing Performance Evaluation.**

I. Introduction

As network services and Internet applications evolve, network traffic is becoming increasingly complex and dynamic. The convergence of data, telephony and television services on an all-IP network directly translates into a much higher variability and complexity of the traffic injected into the network. Recent Internet traffic studies from major network technology vendors like Cisco Systems forecast the advent of the Exabyte era [1], a massive increase in network traffic driven by high-definition video. Furthermore, current evolution and deployment-rate of broadband access technologies (e.g. Fiber To The Home technology) is such that the old assumption of infinitely provisioned core links will soon become obsolete; market research reports like [2] forecast a value of bandwidth demand per user as high as 50 Gb/sec in 2030. In this context, simply upgrading link capacities may no longer be an economically viable solution to cope with dynamic traffic. To make matters worse, the presence of unexpected events such as network equipment failures, large-volume network attacks, flash crowd occurrences and even external routing modifications induces large uncertainty in traffic patterns. In the light of this traffic scenario, we study the problem of intradomain routing optimization under traffic uncertainty. This uncertainty is assumed to be an exogenous traffic modification, meaning that traffic variations are not produced within the domain for which routing is optimized but are due to external and difficult to predict events.

Recent works [8]–[11] have proposed a plausible solution to the routing optimization under traffic uncertainty problem: the Robust Routing (RR) approach. In RR, traffic uncertainty is taken into account directly within the routing optimization, computing a single routing configuration for all traffic demands within some *uncertainty set* where traffic is assumed to vary. This uncertainty set can be defined in different ways, depending on the available information: largest values of links load previously seen, a set of previously observed traffic demands (previous day, same day of the previous week), etc. The criteria to search for this unique routing configuration in these works is to minimize the maximum link utilization (i.e., the utilization of the most loaded link in the network) for all traffic demands of the corresponding uncertainty set. While this routing configuration is not optimal for any single traffic demand within the set, it minimizes the worst case performance over the whole set.

The RR approach can be used as a preemptive TE technique to deal with dynamic and uncertain traffic demands. It can handle unexpected traffic variations with relatively low performance loss, depending on the size of the uncertainty set. However, RR presents some conception problems and serious shortcomings in its current state which we highlight and try to ease in this work. The first drawback of current RR is related to the objective function it intends to minimize. Optimization under uncertainty is generally more complex than classical optimization, which forces the use of simpler optimization criteria such as maximum link utilization (MLU). The MLU is not the most suitable network-wide optimization criterion; setting the focus too strictly on MLU often leads to worse distributions of traffic, adversely affecting the mean network load and thus the total network end-to-end delay, an important QoS indicator. It is easy to see that the minimization of the MLU in a network topology with heterogeneous link capacities may lead to poor results as regards global network performance. The second drawback of RR we identify is its inherent dependence on the definition of the uncertainty set of traffic demands: larger sets allow to handle a broader group of traffic demands, but at the cost of routing inefficiency; conversely, tighter sets produce more efficient routing schemes, but subject to poor performance guarantees. Thus, considering a unique RR configuration to address both traffic in normal operation

and unexpected traffic variations is an inefficient strategy: a single routing can not be suitable for both situations.

A. Related Work

There is a large literature on routing optimization with uncertain traffic demands. Traditional algorithms rely on a single or a small group of expected traffic demands to compute optimal and reliable routing configurations. An extreme case is presented in [5], where routing is optimized for a single estimated traffic demand and is then applied for daily routing. Traffic uncertainty is characterized by multiple traffic demands in [6] (set of traffic demands from previous day, same day of previous week, etc.), where different mechanisms to find optimal routes for the set are presented. In the previously described scenario this perspective is no longer suitable (see [7] for further arguments in this sense).

A different approach has emerged in the recent years to cope with the traffic increasing dynamism and the need for cost-effective solutions, Dynamic Load-Balancing (DLB) [13]–[16]. In DLB, traffic is split among a priori established paths in order to avoid network congestion. The two most well-known proposals in this area are MATE and TeXCP. In MATE [13], a convex link cost function is defined, which depends on the link capacity and the link load. The objective is to minimize the total network cost, for which a simple gradient descent method is proposed. TeXCP [14] proposes a somewhat simpler objective: minimize the biggest utilization each traffic demand obtains in its paths. Another DLB scheme which has the same objective but a relatively different mechanism is REPLEX [15]. In [16], we use a link cost function based on measurements of the queueing delay, which results in better global performance from a QoS perspective. DLB presents a desirable property, that of keeping routing adapted to dynamic traffic. However, DLB algorithms present a trade-off between adaptability and stability which might be particularly difficulty to address under significant and abrupt traffic changes. Besides, network operators are reluctant to use dynamic mechanisms and prefer stable routing configurations, as they claim they get a better feeling of what is going on in the network.

The last category of algorithms consists of Robust Routing techniques [8]–[12]. The objective in RR is to find a unique static routing configuration that fulfills a certain criterion for a broad set of traffic demands, generally the one that minimizes the maximum link utilization over the whole set of demands. In [8], authors capture traffic variations by introducing a polyhedral set of demands, which allows for easier and faster linear optimization. [10] applies this robust technique to compute a robust MPLS routing configuration without depending on traffic demand estimation, and discusses corresponding methods for robust OSPF optimization. Oblivious Routing [9] also defines linear algorithms to optimize worst-case MLU for different sizes of traffic uncertainty sets. [12] analyses the use of robust routing through a combination of traffic estimation techniques and its corresponding estimation error bounds, in order to shrink the set of traffic demands. In [11] authors introduce COPE, a RR mechanisms that optimizes routing for predicted demands and bounds worst-case MLU to ensure acceptable efficiency under unexpected traffic events. As we mentioned before, RR presents two important shortcomings, the former related to the objective function it intends to minimize (i.e., MLU minimization) and the latter as an inherent consequence of its stability property (i.e., using a single routing configuration for all traffic events).

B. Contributions of the Paper

In this paper we propose and evaluate new variants of the RR approach to alleviate the two problems identified in current proposals. As regards the objective function to minimize, we firstly propose to minimize the mean link utilization instead of the MLU. The mean link utilization provides a better image of network-wide performance, as it does not depend on the particular load or capacity of each single link in the network but on the average value. A direct minimization of the mean link utilization does not assure a bounded MLU, which is not practical from an operational point of view. Thus, we minimize the mean link utilization while bounding the MLU by a certain utilization threshold a priori defined. This adds a new difficult to set constraint to the problem, namely how to define this utilization threshold. We further improve our proposal by providing a multiple objective optimization criterion, where both the MLU and the mean link utilization are minimized simultaneously. We evaluate the improvements of our proposals from a QoS perspective, using the mean path end-to-end queuing delay as a measure of global performance.

Regarding the trade-off between routing performance and routing reliability, the only previous work that has evidenced the problem and proposed some solution is COPE [11]. Nevertheless, COPE proposes a single routing configuration to handle expected as well as large and abrupt traffic variations, which is clearly not the best solution. In [17] we have recently proposed a solution to manage this trade-off, known as Reactive Robust Routing (RRR). Basically, RRR consists of constructing a RR configuration for expected traffic in nominal operation, adapting this nominal routing configuration after the detection and localization of a large and long-lived traffic modification. RRR provides good performance for both nominal operation and unexpected traffic, but it is difficult to deploy in a real implementation, because of the routing reconfiguration step. Reconfiguring the routing of an entire Autonomous System is a nontrivial task. In this paper we modify the RRR approach, using a preemptive Load Balancing algorithm to balance traffic among prestablished paths after the localization of a large volume traffic modification.

The remainder of this paper is organized as follows. In section II we recall the traditional Robust Routing approach, exposing the above mentioned problems with some real evaluations. Section III presents the proposed variants to the former RR mechanism, designed to improve the detected shortcomings. The evaluation of the proposed algorithms under different traffic scenarios and the discussion of results are provided in section IV. Finally, section V concludes this work and presents some future perspectives.

II. Stable Robust Routing

Let us begin by introducing the notation used in this paper. The network topology is defined by n nodes and a set $L = \{l_1, \ldots, l_q\}$ of q links, each with a corresponding capacity $c_i,\ i = 1, \ldots, q$. The Traffic Matrix (TM) $X = \{x_{i,j}\}$ denotes the traffic demand between every origin node i and every destination node j ($i \neq j$) of the network; we shall note each of these origin-destination pairs as OD pairs, and each origin-destination traffic demand $x_{i,j}$ as OD flows. Let $X = \{x_k\}$ be the vector representation of the TM, where we have reordered OD flows by index $k = 1, \ldots, m$ $(m = n.(n-1))$. Let $N = \{\text{OD}_1, \ldots, \text{OD}_m\}$ be the set of m OD pairs. We consider a multi-path network topology, where each OD flow x_k can be arbitrarily split among a set of p_k origin-destinations paths P_k. In this sense, we shall call r_p^k the portion of traffic flow x_k sent through path $p \in P_k$, where $0 \leqslant r_p^k \leqslant 1$ and $\sum_{p \in P_k} r_p^k = 1$.

Let λ_l^p be an indicator variable that takes value 1 if path p traverses link l and 0 otherwise, and $Y = \{\rho_1, \ldots, \rho_q\}$ a vector representation of links traffic load. Then X and Y are related through the routing matrix R, a $q \times m$ matrix $R = \{r_l^k\}$ where $r_l^k = \sum_{p \in P_k} \lambda_l^p \,.\, r_p^k$. The variable r_l^k indicates the fraction of OD flow x_k routed through link l; this results in the following relation:

$$Y = R \,.\, X \tag{1}$$

Given X, the multi-path routing optimization problem consists in choosing the set of paths P_k for each OD pair k and computing the routing matrix R, in order to optimize a certain objective function $f(X, R)$. A simplified version of this problem is the optimal load-balancing problem which, *given* a set of paths, calculates R. The most popular TE objective function $f(X, R)$ has traditionally been the maximum link utilization u_{max}, defined as:

$$u_{\text{max}}\,(X, R) = \max_{l \in L} \{u_l\} \tag{2}$$

where $u_l = \rho_l / c_l$ stands for the link utilization; a value of u_l close to 1 indicates that the link is operating near its capacity. Network operators usually prefer to keep links utilization relatively low in order to support sudden traffic increases and link/node failures.

Finding a multi-path routing configuration minimizing u_{max} is an instance of the classical multi-commodity flow problem which can be formulated as a simple linear program [3]. For a single known traffic matrix X, the problem can be easily solved by linear programming techniques [4]. However, as we have previously discussed, traffic demands are uncertain and difficult to predict in current scenario, and all we can expect is to find them within some bounded uncertainty set.

In a robust perspective of the multi-path routing optimization problem, demand uncertainty is taken into account within the routing optimization, computing a single routing configuration for all demands within some uncertainty set. In this work we consider a polyhedral uncertainty set $\mathbb{X}$, more precisely a *polytope* as in [8], based on the intersection of several half-spaces that result from linear constraints imposed to traffic demand.

$$\begin{aligned} &\text{minimize} && u_{\text{max}} && && (3)\\ &\text{subject to:} \\ &\sum_{k \in N} \sum_{p \in P_k} \lambda_l^p \,.\, r_p^k \,.\, x(k) \;\leqslant\; u_{\text{max}}.c_l && && \forall\, l \in L,\ \forall\, X \in \mathbb{X} \\ &\sum_{p \in P(k)} r_p^k \;=\; 1 && && \forall\, k \in N \\ &r_p^k \;\geqslant\; 0 && && \forall\, p \in P_k,\ \forall\, k \in N \\ &u_{\text{max}} \;\leqslant\; 1 \end{aligned}$$

As an example, let us define an uncertainty set $\mathbb{X}$ based on a given routing matrix R_{o} and the peak-hour links traffic load Y^{peak} obtained with this routing matrix:

$$\mathbb{X} \;=\; \left\{X \in \mathbb{R}^m, R_{\text{o}}.X \leqslant Y^{\text{peak}},\ X \geqslant 0\right\}$$

Observe that this definition of the uncertainty set has a major advantage: routing optimization can be performed from easily available links traffic load Y without even knowing the actual value of the traffic demand X. Figure 1 depicts the obtained uncertainty set, based on the convex intersection of q half-spaces of the form $\mathbf{r}_{\text{o}}^i \cdot X \leqslant \rho_i^{\text{peak}}$, $\forall i \in L$, where $\mathbf{r}_{\text{o}}^i$ stands for the i-th row of the routing matrix R_{o}.

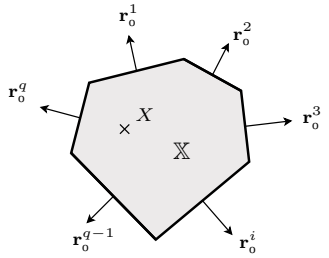


Fig. 1. The uncertainty set $\mathbb{X}$ as a polytope.

The traditional Robust Routing Optimization Problem (RROP) defined in (3) consists of minimizing the maximum link utilization u_{max}, considering all demands within $\mathbb{X}$. The solution to the problem is twofold: on the one hand, a routing configuration R_{robust}, and on the other hand, a worst-case performance threshold $u_{\text{max}}^{\text{robust}}$:

$$\begin{aligned} R_{\text{robust}} &= \operatorname*{argmin}_{R} \max_{X \in \mathbb{X}} u_{\text{max}}(X, R) \\ u_{\text{max}}^{\text{robust}} &= \max_{X \in \mathbb{X}} u_{\text{max}}(X, R) \end{aligned}$$

Given a suitable definition of the uncertainty set, the obtained robust routing configuration R_{robust} is applied during long periods of time; in this sense, we refer to Robust Routing as *Stable Robust Routing* (SRR). The authors of [8] have shown that the RROP can be efficiently solved by linear programming techniques, applying a combined columns and constraints generation method. This method iteratively solves the problem, progressively adding new constraints and new columns to the problem. The new constraints are the extreme points of the uncertainty set $\mathbb{X}$, and the new columns represent new paths added to reduce the objective function value. Only extreme points of $\mathbb{X}$ are added as new constraints, as it is easy to see that every traffic demand $X \in \mathbb{X}$ can be expressed

as a linear combination of these extreme demands. Regarding new added paths, the algorithm in [8] may not be the best choice from a practical point of view since the number of paths for each OD pair is not a priori restricted and the characteristics of added paths are not controlled. For example, it would be interesting to have disjoint paths to route traffic from each single OD pair, improving resilience. For this reason we modify the algorithm to select new paths, both limiting the maximum number of paths in P_k and taking as new candidates the shortest paths with respect to link weights w_l^i:

$$w_l^i = \frac{1}{\epsilon + \left(1 - {r_l^k}^i\right)} \tag{4}$$

where ${r_l^k}^i$ corresponds to the fraction of traffic flow x_k that traverses link l after iteration i and ϵ is a small constant that avoids numerical problems. If OD pair k uses a single path p at iteration i, ${r_l^k}^i = 1$ for every link $l \in p$, and so this path is removed from the graph where new shortest paths are computed ($w_l \to \infty$, $\forall l \in p$). While this may result in a sub-optimal performance, it allows a real and practical implementation. In case there are no disjoint paths for OD pair k, we use the column constraint generation method used in [8] to add new paths for OD pair k.

A. Shortcomings of Stable Robust Routing: An Example

In this section we shall present some simulations that will help us gain insight into the Stable Robust Routing mechanism and highlight the previously discussed shortcomings. We use the Abilene Network as the environment for simulations. Abilene is a high-speed Internet2 backbone network, connecting 12 router-level nodes through 30 optical links (we only consider intra-domain links). The used router-level network topology and traffic demands are available at [19]. Traffic data consists of 6-month traffic matrices collected every 5' via Netflow from the Abilene Observatory [20]. As measured traffic demands do not significantly load the network, we re-scaled them by multiplying all their entries by a constant. The dataset in [19] also provides the static routing configuration R_o deployed in Abilene during 6-month TMs measurement campaign.

Let us discuss the issue related to the objective function used in the traditional SRR algorithm. As we stated in the Introduction, the maximum link utilization is a local performance indicator, and a routing configuration minimizing $u_{\max}$ may often lead to a worse distribution of traffic, adversely affecting the global performance of the network. Besides, while it is true that overloaded links tend to cause QoS degradation (e.g., larger delays and packet losses, throughput reduction, etc.), $u_{\max}$ does not represent a direct QoS indicator, a desirable property in the context of QoS provisioning.

In order to evaluate SRR from a network-wide QoS perspective, let us consider a performance indicator directly related to QoS: the path end-to-end (e2e) delay. The e2e delay on a path is the sum of the delays on each link of the path. The delay on each link consists of two components, namely the queuing delay (i.e., buffer and service delay) and the link propagation delay. The former depends on the link load, while the latter is constant. In this sense and as a simplification to the problem, we shall consider the e2e path queuing delay as a measure of performance. Assume that queuing delay on link l is given by the function $d_l(\rho_l)$. Given this function, we can compute the e2e queuing delay of path p as $d_p = \sum_{l \in p} d_l(\rho_l)$. In order to evaluate the network-wide performance of SRR, we define the expected e2e path queuing delay d_{mean} as follows:

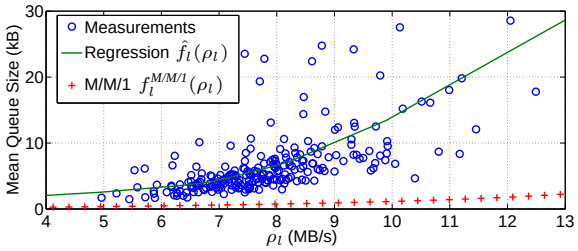


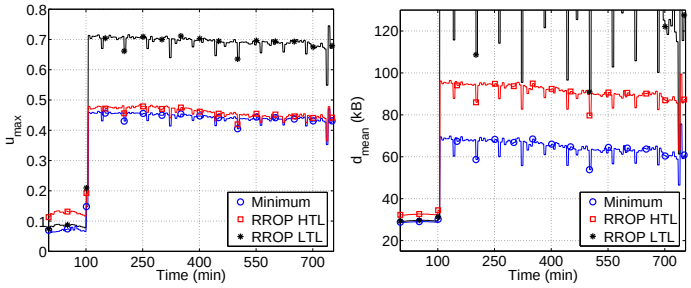
Fig. 2. Mean queue size, measurements and approximations

$$d_{\text{mean}}(X, R) = \sum_{k \in N} \sum_{p \in P_k} \left(r_p^k \,.\, x_k\right) d_p = \sum_{l \in L} \rho_l \,.\, d_l(\rho_l) \tag{5}$$

That is to say, a weighted mean e2e queuing delay, where the weight for each path is how much traffic is sent through it ($r_p^k \,.\, x_k$), or in terms of links, the weight for each link is how much traffic is traversing it (ρ_l). A large mean e2e queuing delay translates into bad performance for all the traffic and not only for the traffic that traverses a particular loaded link. We prefer a weighted mean queuing delay to a simple total delay because it reflects more precisely performance as perceived by traffic. Two situations where the total delay is the same, but in one of them most of the traffic is traversing heavily delayed links should not be considered as equivalent. Note that, by Little's law, the value $f_l(\rho_l) = \rho_l \,.\, d_l(\rho_l)$ is proportional to the volume of data in the queue of link l. We will then use this last value as the addend in the last sum in (5), since it is easier to measure than the queuing delay.

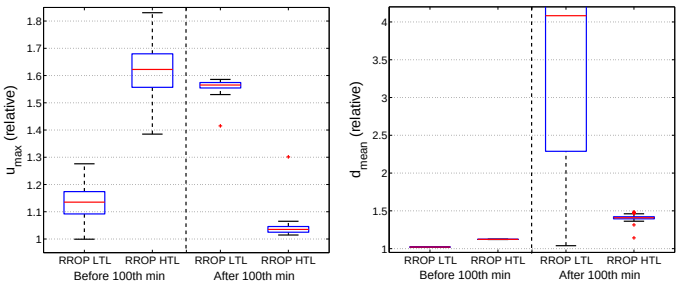
The function $f_l(\rho_l)$ is unknown and in the literature it is generally estimated using a classical *M/M/1* model, where $f_l^{M/M/1}(\rho_l) = \rho_l/(c_l - \rho_l)$ [18]. However, in [16] we show that a simple *M/M/1* model has little to do with reality, and so we propose to use a non-parametric regression technique to estimate $f_l(\rho_l)$ from measurements without assuming any given model. Figure 2 depicts the real mean queue size of an operational network link at Tokyo obtained from [21], together with the *M/M/1* estimation $f_l^{M/M/1}(\rho_l)$ and the non-parametric regression $\hat{f}_l(\rho_l)$. It is clear that $f_l^{M/M/1}(\rho_l)$ consistently underestimates the real queue size value, while $\hat{f}_l(\rho_l)$ provides quite accurate results.

Let us evaluate the performance of SRR as regards both $u_{\max}$ and d_{mean}. From now on we shall use RROP as a reference to SRR, recalling that the robust routing optimization problem is the one described in (3). In this evaluation we consider a traffic scenario that presents and abrupt and large volume increase



(a) Maximum link utilization (b) Mean e2e queuing delay

Fig. 3. Maximum link utilization and mean end-to-end queuing delay. Traffic demand volume abruptly increases after the 100th minute.



(a) Relative maximum link utilization (b) Relative e2e queuing delay

Fig. 4. Maximum link utilization and mean end-to-end queuing delay, boxplot performance summary. Depicted results are relative to the optimal values.

due to an external routing modification. This corresponds to the TMs with indexes between 1050 and 1200 from dataset X23 in [19]. Figure 3 depicts the described traffic scenario. The evaluation starts with a normal low traffic load situation, but after the 100th minute one of the OD flows abruptly increases its traffic volume, loading the links it traverses until the end of the evaluation. Based on the static routing matrix of Abilene R_o we define two different polytopes, the former adapted to the Low Traffic Load period (LTL period, before the 100th minute) and the latter adapted to the High Traffic Load period (HTL period, after the 100th minute):

$$\mathbb{X}^{\text{LTL}} = \{X \in \mathbb{R}^m, R_o.X \leqslant Y^{\text{LTL}},\ X \geqslant 0\}$$

$$\mathbb{X}^{\text{HTL}} = \{X \in \mathbb{R}^m, R_o.X \leqslant Y^{\text{HTL}},\ X \geqslant 0\}$$

We assume that traffic is known in advance in both definitions, and take Y^{LTL} and Y^{HTL} as the maximum link load values observed during the LTL and HTL periods respectively. We compute two different robust routing configurations for both polytopes; RROP LTL corresponds to the SRR configuration for polytope $\mathbb{X}^{\text{LTL}}$, and RROP HTL for polytope $\mathbb{X}^{\text{HTL}}$. In this evaluation, both RROP LTL and RROP HTL use the same set of paths, namely the paths obtained from (3) for polytope $\mathbb{X}^{\text{LTL}}$. Solving (3) for a given set of paths consists of only adding new extreme points of polytope $\mathbb{X}$ (i.e., only new constraints are added).

Figure 3 depicts (a) the maximum link utilization u_{max} and (b) the mean end-to-end queuing delay d_{mean} during the evaluation period. Both u_{max} and d_{mean} are updated every 5 minutes, when a new TM is measured. As a reference for comparison, we also compute the minimum values $u^{\text{opt}}_{\text{max}}(X)$ and $d^{\text{opt}}_{\text{mean}}(X)$ for every single TM X of the evaluation period, using once again the same set of paths. $u^{\text{opt}}_{\text{max}}(X)$ is computed with a simplified version of (3), where there is only one TM X instead of a set $\mathbb{X}$. $d^{\text{opt}}_{\text{mean}}(X)$ is computed using the algorithms in [16]. Figure 4 presents a boxplot summary of the performance of RROP LTL and RROP HTL relative to the optimal values $u^{\text{opt}}_{\text{max}}(X)$ and $d^{\text{opt}}_{\text{mean}}(X)$, for both LTL and HTL periods.

Let us first focus the attention on the performance of RROP HTL after the 100th minute. Despite achieving an almost optimal performance as regards u_{max} (a difference smaller than 4%), RROP HTL obtains a queuing delay that constantly exceeds the optimum by almost 40% under a fairly network load. Such a difference may not be even acceptable from a QoS perspective, where end-to-end delays are even more important than network congestion. As we will show latter, this loss in performance is a direct consequence of the local criterion used in RROP.

The second interesting observation comes from the difference between RROP LTH and RROP LTL performances before and after the abrupt traffic volume increase; figure 3(a) shows that, despite an almost negligible network load, RROP LTL outperforms RROP HTL by almost 50% of relative utilization during the LTL period, while the opposite happens during the HTL period. The difference is not that big as regards delay before the 100th minute, but it becomes really important after the volume increase, where RROP LTL obtains a really bad performance. These results are somehow expected given the polytopes definition, and evidences both the dependence of RROP on the uncertainty set definition and the inherent consequence of using a single static routing configuration under large traffic variations. A final remark about this simple evaluation and the definition of the uncertainty set; we have considered that traffic was known in advance for the definition of both polytopes $\mathbb{X}^{\text{LTL}}$ and $\mathbb{X}^{\text{HTL}}$. While traffic during the LTL period is easy to predict, the definition of $\mathbb{X}^{\text{HTL}}$ in a real traffic scenario is a challenging task. We will come back to this issue in the following section.

III. New Mechanisms for Robust Routing

In this section we shall present three enhanced mechanisms to overcome the problems of SRR evidenced in the previous evaluation. We will first introduce and evaluate two similar mechanisms to attain better global performance as regards end-to-end delay. Then we will present a mechanism to manage the problem of defining the uncertainty set under unexpected and large traffic variations, previously introduced in [17]. In this work we provide some slight modifications to the former algorithm which allow a real deployment of the proposal.

$$
\begin{aligned}
&\text{minimize} && u_{\text{mean}} && && (7)\\
&\text{subject to:} && && &\\
&\sum_{l\in L}\sum_{k\in N}\sum_{p\in P_k}\tfrac{1}{c_l}\lambda_l^p\,.\,r_p^k\,.\,x(k) && \leqslant && u_{\text{mean}}.q && \forall\, X\in\mathbb{X}\\
&\sum_{k\in N}\sum_{p\in P_k}\lambda_l^p\,.\,r_p^k\,.\,x(k) && \leqslant && u_{\text{max}}^{\text{thres}}.c_l && \forall\, l\in L,\ \forall\, X\in\mathbb{X}\\
&\sum_{p\in P(k)} r_p^k && = && 1 && \forall\, k\in N\\
&r_p^k && \geqslant && 0 && \forall\, p\in P_k,\ \forall\, k\in N
\end{aligned}
$$

$$
\begin{aligned}
&\text{minimize} && u_{\text{aof}}=\beta\,.\,u_{\text{max}}+(1-\beta)\,.\,u_{\text{mean}} && && (8)\\
&\text{subject to:} && && &\\
&\sum_{l\in L}\sum_{k\in N}\sum_{p\in P_k}\tfrac{1}{c_l}\lambda_l^p\,.\,r_p^k\,.\,x(k) && \leqslant && u_{\text{mean}}.q && \forall\, X\in\mathbb{X}\\
&\sum_{k\in N}\sum_{p\in P_k}\lambda_l^p\,.\,r_p^k\,.\,x(k) && \leqslant && u_{\text{max}}.c_l && \forall\, l\in L,\ \forall\, X\in\mathbb{X}\\
&\sum_{p\in P(k)} r_p^k && = && 1 && \forall\, k\in N\\
&r_p^k && \geqslant && 0 && \forall\, p\in P_k,\ \forall\, k\in N
\end{aligned}
$$

A. Improving Network-Wide Performance

As we showed in figure 3(b), the minimization of u_{max} leads to a distribution of traffic that results in an excessive end-to-end delay. Using the mean delay $d_{\text{mean}}\,(X,R)$ as the objective function in (3) would be an interesting approach to ease the problem; however, $f_l(\rho_l)$ is a non-linear function and the optimization problem becomes too difficult to solve. As we previously said, optimization under uncertainty is more complex than classical optimization and simple optimization criteria should be used. Let us consider a very simple network-wide linear objective function, namely the mean link utilization $u_{\text{mean}}\,(X,R)$, defined as:

$$u_{\text{mean}}\,(X,R)=\frac{1}{q}\sum_{l\in L}u_l \tag{6}$$

The mean link utilization considers at the same time the load of every link in the network and not only the utilization of the most loaded link; as we will show in the results, such an objective function provides a better global performance as regards end-to-end delay. However, a direct minimization of u_{mean} does not assure a bounded maximum link utilization, which is not practical from an operational point of view. In this sense, we propose to change the objective function in (3) by u_{mean}, while bounding the maximum link utilization by a certain threshold $u_{\text{max}}^{\text{thres}}$ a priori defined. This results in the Robust Routing Mean Utilization Optimization Problem (RRMP) defined in (7).

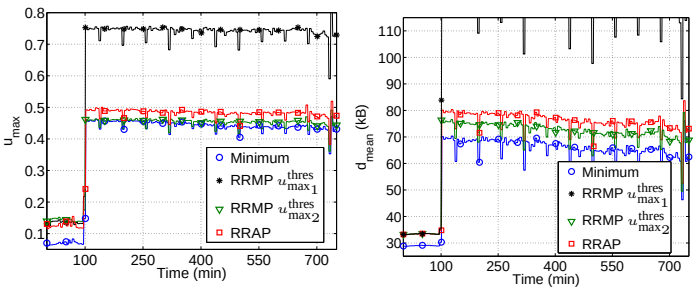
Problem (7) is solved in the same way as (3), using the same recursive algorithm proposed in [8]. Note that (7) adds only a new constraint per each new traffic demand in $\mathbb{X}$ (in fact, for each extreme point of $\mathbb{X}$). The drawback of (7) is on its dependence on the value of $u_{\text{max}}^{\text{thres}}$, which directly influences the routing performance as we will shortly see. An interesting choice for $u_{\text{max}}^{\text{thres}}$ would be to use the output of (3), namely $u_{\text{max}}^{\text{robust}}$. To some extent this would result in a similar routing solution but with better traffic balancing.

A alternative approach would be to minimize both the value of u_{max} and u_{mean} at the same time, what constitutes a problem of multi-objective optimization. The problem with multi-objective optimization is that traditional single-objective optimization techniques can not be directly applied. An intuitive and easy approach to solve a multi-objective optimization problem using single-objective optimization techniques is to construct a single aggregated objective function (AOF) that combines both objective functions. We define a weighted linear combination of u_{max} and u_{mean} as the new objective function $u_{\text{aof}}=\beta\,.\,u_{\text{max}}+(1-\beta)\,.\,u_{\text{mean}}$, where $0\leqslant\beta\leqslant 1$ is the combination fraction. Despite its simple form, this new objective is very effective and provides accurate results for both performance indicators. The new obtained optimization problem is the Robust Routing AOF Optimization Problem (RRAP), defined in (8). Once again, problem (8) is solved with the same algorithms used in (3).

B. Comparison between RRMP and RRAP

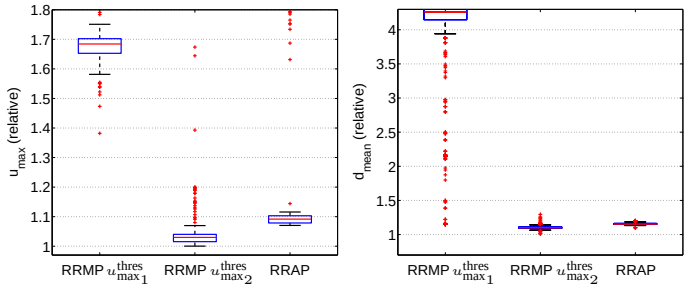
We will now evaluate both the RRMP and RRAP versions of SRR in the same traffic scenario previously used in section II-A. In order to appreciate the dependence of RRMP on the maximum link utilization threshold $u_{\text{max}}^{\text{thres}}$, two different thresholds are used in the evaluation: $u_{\text{max}_1}^{\text{thres}}=1$, which corresponds to the constraint $u_{\text{max}}\leqslant 1$ in (3), and $u_{\text{max}_2}^{\text{thres}}=u_{\text{max}}^{\text{robust}}$, where $u_{\text{max}}^{\text{robust}}$ is the output of RROP HTL in section II-A. In the case of RRAP, the weight β is set to 0.5, namely an even balance between u_{max} and u_{mean}. Figures 5 and 6 depicts the comparison as regards (a) maximum link utilization and (b) mean end-to-end queuing delay. Let us focus the attention on the operation after the 100th minute, as all robust routing configurations use $\mathbb{X}^{\text{HTL}}$ as uncertainty set. To be as fair as possible, both RRMP and RRAP use the same set of paths as those used by RROP in figure 3. RRMP performance clearly depends on the threshold $u_{\text{max}}^{\text{thres}}$; in the first case, the attained maximum link utilization is well beyond the optimal values, reaching almost a 70% of relative performance degradation. This overload directly translates into huge mean end-to-end queuing delays. Results are quite impressive when considering the second threshold, both as regards u_{max} and d_{mean}. RRMP $u_{\text{max}_2}^{\text{thres}}$ provides a highly efficient robust routing configuration, showing that it is possible to improve current implementations of SRR with a slight modification of the objective function. However, this dependence on the threshold $u_{\text{max}}^{\text{thres}}$ introduces a new tunable parameter, something undesirable when looking for solutions that simplify network management.

As regards RRAP, obtained results are slightly worse than those obtained by RRMP $u_{\text{max}_2}^{\text{thres}}$, but still very close to the optimal performance, with a relative performance degradation of about 10% as regards u_{max} and d_{mean} w.r.t. an optimal routing configuration. Nevertheless, RRAP has no tunable parameter apart from the combination factor β, which in fact is set to a half independently of the traffic situation.



(a) Maximum link utilization (b) Mean e2e queuing delay

Fig. 5. Maximum link utilization and mean end-to-end queuing delay for RRMP and RRAP.



(a) Relative maximum link utilization (b) Relative e2e queuing delay

Fig. 6. Maximum link utilization and mean end-to-end queuing delay for RRMP and RRAP, boxplot performance summary. Depicted results are relative to the optimal values.

C. The Reactive Robust Routing

As we showed in section II-A, the definition of the uncertainty set has a major impact on the performance of SRR. In particular, we saw that using a single definition of uncertainty set under highly variable traffic can not provide routing efficiency for both normal operation traffic and unexpected traffic events. Despite being one of its most important features, using a single SRR configuration is not the best strategy.

In [17] we proposed an adaptive version of SRR, known as the Reactive Robust Routing (RRR). The basic idea in RRR consists of computing a primal robust routing configuration $R^{\text{o}}_{\text{robust}}$ for expected traffic variations in normal operation within a primal polytope $\mathbb{X}_{\text{o}}$. This polytope is defined as in section II, based on a certain fixed routing configuration R_{o} and the expected links traffic load we shall call $Y_{\text{o}} = \{\rho_{\text{o}_i}\}$. Additionally, a set of m *anomaly* polytopes $\mathbb{X}_j$ are defined, and a preemptive robust routing configuration R^{j}_{robust} is computed for each of these anomaly polytopes.

Let us explain the concept of an anomaly polytope. In figure 3, the abrupt increase in traffic volume is caused by a single *anomalous* OD flow x_k that unexpectedly carries a many times bigger traffic load θ due to an external routing modification. After this exogenous unexpected event, the normal operation traffic demand X takes the value $X' = X + \theta.\boldsymbol{\delta}_k$, where $\boldsymbol{\delta}_k = (\delta_{1,k}, \ldots, \delta_{k,k}, \ldots, \delta_{m,k})^T$, $\delta_{i,k} = 0$ if $i \neq k$ and $\delta_{k,k} = 1$. We shall designate this unexpected traffic increase in OD flow x_k as anomalous traffic event A_k. The anomaly

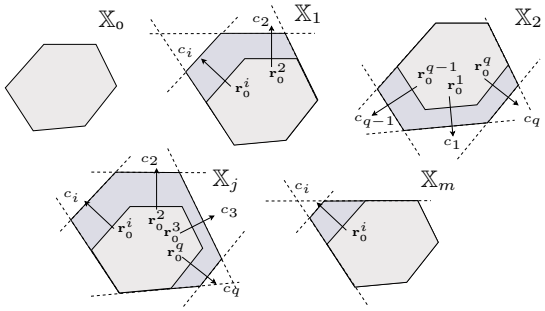


Fig. 7. Different anomaly polytopes for preemptive robust routing computation.

polytope $\mathbb{X}_k$ results from expanding the primal polytope $\mathbb{X}_{\text{o}}$ in the directions of the links that traverses the anomalous OD flow x_k, with respect to R_{o}. The reader should bear in mind that the kind of unexpected traffic events we deal with are independent of the intradomain routing; these events originate outside the network and propagate between origin-destination nodes. This justifies the relevance of the polytope expansion with respect to R_{o}. The obtained polytope $\mathbb{X}_k$ is the smallest polytope that contains the unexpected traffic demand X' and thus, the corresponding robust routing configuration R^{k}_{robust} provides accurate performance under its occurrence. Figure 7 explains the idea of the multiple anomaly polytope expansion. As before, $\mathbf{r}^i_{\text{o}}$ stands for the i-th row of the routing matrix R_{o}.

Note that in a real scenario it is not possible to predict the size of the anomalous traffic θ. As a consequence, the primal polytope $\mathbb{X}_{\text{o}}$ is expanded to the limits of link capacities, obtaining the following anomaly polytope for each anomalous traffic event A_k:

$$\mathbb{X}_k = \left\{ X \in \mathbb{R}^m, R_{\text{o}}.X \leqslant Y^{A_k},\ X \geqslant 0 \right\},\ \forall k \in N \qquad (9)$$

In (9), the i-th component of Y^{A_k} takes the value ρ_{o_i} if $r^{i,k}_{\text{o}} = 0$, or the value c_i if $r^{i,k}_{\text{o}} > 0$, being $r^{i,k}_{\text{o}}$ the element (i,k) of R_{o}.

Given the primal and the m preemptive robust routing configurations $R^{\text{o}}_{\text{robust}}$ and R^{j}_{robust}, RRR uses different anomaly detection/localization sequential algorithms to detect the occurrence of an anomalous event A_k, switching routing from $R^{\text{o}}_{\text{robust}}$ to R^{k}_{robust}. In particular, RRR defines a recursive anomaly detection function we shall call $g(t)$, and m recursive anomaly localization functions $s_j(t), j \in N$. These functions are updated with every new traffic measurement at time t, and when the difference between $g(t)$ and $g(t-1)$ exceeds a certain anomaly detection threshold $\Delta_{\text{detection}_1}$ an anomaly is declared. Functions $s_j(t)$ are used to locate the anomalous OD flow; under the presence of an anomaly in OD flow x_k at time t, the function $s_k(t)$ dramatically increases its value w.r.t. the rest of the $s_j(t)_{j \neq k}$ functions, highlighting the anomalous OD flow. Additionally, RRR uses a similar detection algorithm to detect the end of the anomaly at OD flow x_k, switching routing from R^{j}_{robust} to $R^{\text{o}}_{\text{robust}}$ when normal operation is regained. We refer the reader to [17] for additional details on the implementation of RRR.

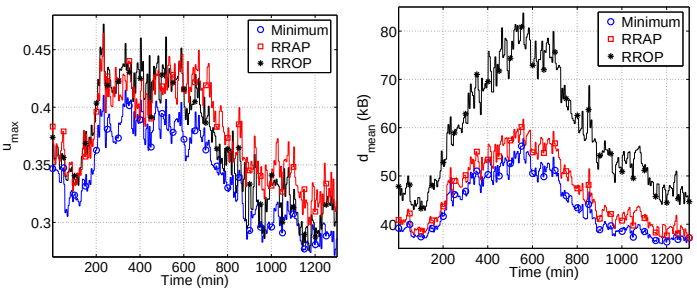
RRR can handle large and unexpected traffic variations in single OD flows quite effectively (the case of multiple simultaneous anomalies is beyond the scope of RRR). However, given the difficulty involved in modifying the routing configuration of a large scale network in an on-line fashion, the contributions of RRR are mainly theoretical. This problem can be solved by using a load balancing technique instead of a complete routing reconfiguration. In load balancing, we keep the same set of paths P_k for each OD pair k, and only modify the fractions of traffic sent through each path. Load balancing can be easily performed on-line and does not require any additional modifications in current path-based networks such as MPLS. We shall refer to the load balancing variant of RRR as Reactive Robust Load Balancing (RRLB), stressing the difference between routing reconfiguration and load balancing.

RRLB uses the same set of anomaly polytopes $\mathbb{X}_j$ defined in RRR, but the computation of the m preemptive robust routing configurations R^j_{robust} is slightly modified. The same set of paths P_k obtained during the computation of $R^{\text{o}}_{\text{robust}}$ is used in every R^j_{robust}. As it was done in sections II-A and III-B, routing configurations R^j_{robust} are obtained with a simplified version of the former optimization algorithm, where only new traffic demands are progressively added and no extra paths are created. The following schema gives a high-level description of RRLB:

Algorithm 1 Reactive Robust Load Balancing (RRLB)

compute $R^{\text{o}}_{\text{robust}}$ and P_k, $\forall k \in N$ for primal polytope $\mathbb{X}_{\text{o}}$
using P_k, compute R^j_{robust}, $\forall\, \mathbb{X}_j$, $\forall j \in N$
set anomaly Flag $\phi \leftarrow 0$
for $t = 1{:}\infty$ **do**
  if $\phi = 0$ **then**
    update anomaly detection function $g(t)$
    for $i = 1{:}m$ **do**
      update anomaly localization function $s_i(t)$
    end for
    if $g(t) - g(t-1) > \Delta_{\text{detection}_1}$ **then**
      $j = \underset{j \in N}{\text{argmax}}\, s_j(t)$
      balance traffic according to R^j_{robust}
      $\phi \leftarrow j$
    end if
  else
    update anomaly-end detection function $h_\phi(t)$
    if $h_\phi(t) > \Delta_{\text{detection}_2}$ **then**
      balance traffic according to $R^{\text{o}}_{\text{robust}}$
      $\phi \leftarrow 0$
    end if
  end if
end for

Parameter ϕ is used as an anomaly flag variable that takes value 0 if traffic is in normal operation and value k under the occurrence of an anomalous traffic event A_k. The detection function $h_\phi(t)$ is used to detect the end of the anomalous traffic event in OD flow ϕ. Function $h_\phi(t)$ takes negative values during the presence of an anomaly in OD flow ϕ at time t, increasing its value above a certain detection threshold $\Delta_{\text{detection}_2}$ when the anomaly has ended. Both detection thresholds $\Delta_{\text{detection}_1}$ and $\Delta_{\text{detection}_2}$ are tuned in order to maximize the detection probability while bounding the false alarm rate, see [17] for additional details.



(a) Maximum link utilization (b) Mean e2e queuing delay

Fig. 8. Maximum link utilization and mean end-to-end queuing delay under normal operation. RROP and RRAP are compared against the optimal values.

IV. Evaluation and Discussion

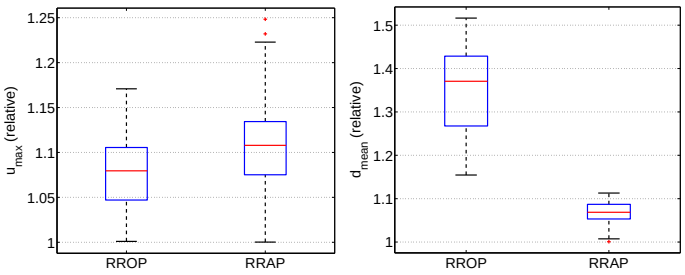
In this section we evaluate the performance of the Robust Routing enhanced mechanisms presented in this work, considering both normal operation and anomalous traffic situations. This allows for performance comparison at different levels of traffic variability. As both RRAP and RRMP provide similar results (when $u^{\text{thres}}_{\text{max}}$ is correctly defined for RRMP), we will only consider the RRAP mechanism in the evaluation.

For the normal operation scenario, we shall compare the efficiency of RRAP against the traditional RROP mechanism. As regards the anomalous traffic scenario, we consider the same situation depicted in figure 3 and compare the execution of RRLB for both RRAP and RROP. In the evaluation, we shall use RRLB-OP and RRLB-AP to designate the Reactive Routing Load Balancing variants of RROP and RRAP respectively.

A. Normal Operation Traffic Scenario

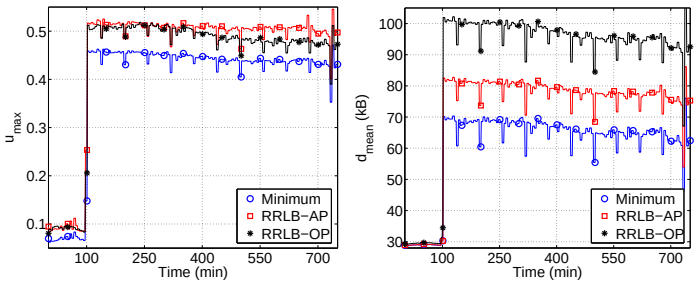
The first case-scenario corresponds to traffic in normal operation. The only variability is due to typical daily fluctuations. Figure 8 presents the evolution of u_{max} and d_{mean} for RROP and RRAP, using a set of 260 TMs from dataset X01 in [19]. Both mechanisms perform similarly as regards maximum link utilization, depicted in figure 8(a). This may be further appreciated in the boxplot summaries presented in figure 9(a), where values are relative to those obtained with an optimal routing configuration. Note that the relative performance degradation is around 10% in both cases.

Figures 8(b) and 9(b) show that results are quite different as regards mean queuing delay. While RRAP has a relative degradation smaller than 10% w.r.t. the optimal delay, RROP systematically obtains an important difference, attaining a performance degradation close to 40%. These results further highlight the limitations of RROP as previously discussed: using u_{max} as a performance objective results in a relatively low maximum utilization, but neglects the rest of the links, impacting the network-wide performance.



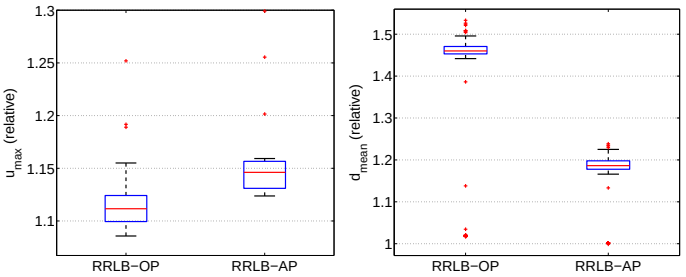
(a) Relative maximum link utilization (b) Relative e2e queuing delay

Fig. 9. Maximum link utilization and mean end-to-end queuing delay under normal operation, boxplot performance summary. Depicted results are relative to the optimal values.



(a) Maximum link utilization (b) Mean e2e queuing delay

Fig. 10. Maximum link utilization and mean end-to-end queuing delay under anomalous traffic. RRLB-OP and RRLB-AP are compared against the optimal values.



(a) Relative maximum link utilization (b) Relative e2e queuing delay

Fig. 11. Maximum link utilization and mean end-to-end queuing delay under anomalous traffic, boxplot performance summary. Depicted results are relative to the optimal values.

B. *Anomalous Traffic Scenario*

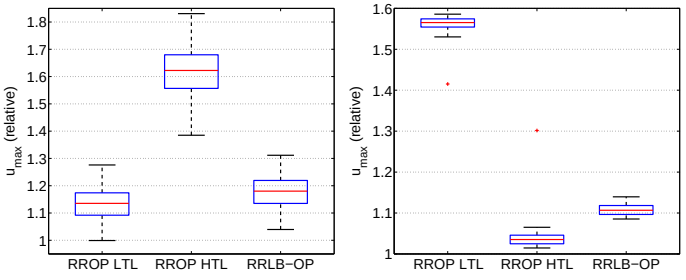
The second case-scenario is the one considered in section II-A, where there is a sudden and abrupt increase of the traffic volume carried by one OD flow. Both RRLB-OP and RRLB-AP use the RRLB mechanism previously described to adapt traffic balancing after the detection of the anomalous traffic variation. As a difference with respect to the evaluation in figure 3, where traffic was assumed known in advance, this case-scenario corresponds to a real situation where traffic anomalies can not be forecast.

Figure 10(a) depicts the attained maximum link utilization before (LTL period) and after the anomalous event (HTL period). Let us first discuss the execution of RRLB, comparing the performance obtained with RRLB-OP against the one obtained by RROP LTL and RROP HTL in figures 3(a) and 4(a). Figure 12 provides a boxplot summary of the relative maximum link utilization (w.r.t. the optimal values) obtained with the 3 algorithms during the (a) LTL period and (b) HTL period. The reader should remember that RROP LTL represents the optimal robust routing configuration for the LTL period as regards RROP, and similarly RROP HTL during the HTL period. It is interesting to see that RRLB-OP obtains a slightly worse performance than the optimum robust routing configurations during the whole evaluation period, with a difference close to 5%. Nevertheless, RRLB-OP represents a real situation where traffic is not assumed known in advance.

Regarding the comparison between RRLB-OP and RRLB-AP, figure 11 shows that both algorithms obtain similar results as regards maximum link utilization, with a relative performance degradation smaller than 15% during the whole evaluation period. Note that while important, this performance degradation is surprisingly small if we consider that traffic increases more than 500% in less than 10 minutes. As regards end-to-end queuing delay, RRLB-AP clearly outperforms RRLB-OP, achieving a relative mean queuing delay almost 30% smaller. These results reinforces once again our observations about the difficulty in RROP to attain global performance, and the advantages of using a simple network-wide objective function in a robust routing algorithm.

V. Conclusions and Future Work

In this work we have studied the problem of intradomain routing optimization under highly variable and difficult to predict traffic demands. We have presented a comprehensive analysis of a plausible solution to the problem, namely the Robust Routing approach, evaluating its performance under different traffic scenarios. From this analysis we have obtained tangible evidence to highlight two important shortcomings of current Robust Routing implementations. On the one hand, we saw that using a local performance criterion such as the maximum link utilization (MLU) does not provide a suitable



(a) Relative $u_{\max}$, LTL period (b) Relative $u_{\max}$, HTL period

Fig. 12. Maximum link utilization for RROP LTL, RROP HTL and RRLB-OP, boxplot performance summary. Depicted results are relative to the optimal values.

objective function as regards network-wide performance and QoS provisioning. In particular, we showed that an almost optimal robust routing configuration w.r.t. MLU can experience rather high mean end-to-end queuing delays, thus impacting on delay sensitive traffic. The maximum link utilization is widely used in current network optimization problems, particularly in most Robust Routing proposals, thus we believe that this simple evidence can help to enhanced future implementations. On the other hand we have shown that using a single routing configuration is not a cost-effective solution when traffic is relatively dynamic. Stable Robust Routing obtains quite poor performance either when faced with non considered traffic demands (tight uncertainty sets) or when designed to manage as many traffic demands as possible (big uncertainty sets). It is clear from our study that some form of dynamism is necessary.

We have proposed solutions to both detected problems of current Robust Routing implementations. To begin with, we have shown that objective optimization functions can be kept simple yet better network-wide performance can be attained. By using a simple combination of performance indicators such as the maximum and the mean link utilization, we obtained a robust routing configuration that definitely outperforms current implementations from a global end-to-end perspective, while achieving almost identical results as regards worst-case link utilization. Additionally, we have extended a previous proposal of our own that correctly manages normal and anomalous traffic situations, improving its execution towards a practical implementation. The Reactive Robust Load Balancing introduces a dynamic approach to deal with unexpected traffic events, balancing load between prestablished paths only when large traffic variations are detected. The evaluations showed than combining this reactive scheme with network-wide performance objectives can offer proper solutions to deal with current and future traffic scenarios.

The framework of Aggregated Objective Functions (AOF) provides interesting results as regards multi-objective optimization, particularly in the context of robust optimization. An AOF approach can be used to construct better objective functions from simple performance indicators and thus deserves further analysis. The estimated queuing delay function provides a difficult to optimize objective, but the problem can be solved provided certain realistic simplifications; this is currently part of our ongoing work.

REFERENCES

[1] Cisco-Systems, “Global IP Traffic Forecast and Methodology 2006-2011, white paper”, 2007 - updated 2008, available at http://www.cisco.com.

[2] KMI Corporation, “Worldwide Markets for Fiberoptics in Broadband Access Networks: Fiberoptic Demand Forecast in Broadband Access Networks”, 2006, available at http://cruonline.crugroup.com/WireandCable/MarketForecasts.

[3] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin “Network Flows: Theory, Algorithms, and Applications”, *Prentice Hall*, 1993.

[4] D. Mitra and K. G. Ramakrishnan, “A Case Study of Multiservice, Multipriority Traffic Engineering Design for Data Netorks”, in *GLOBECOM ’99*, 1999.

[5] M. Roughan, M. Thorup, and Y. Zhang, “Traffic Engineering with Estimated Traffic Matrices”, in *IMC ’03*, 2003.

[6] C. Zhang, Z. Ge, J. Kurose, Y. Liu, and D. Towsley, “On Optimal Routing with Multiple Traffic Matrices”, in *INFOCOM ’05*, 2005.

[7] P. Casas and S. Vaton, “An Adaptive Multi Temporal Approach for Robust Routing”, in *Euro-FGI Workshop on IP QoS and Traffic Control*, 2007.

[8] W. Ben-Ameur and H. Kerivin, “Routing of Uncertain Traffic Demands”, *Optimization and Engineering*, vol. 6, pp. 283-313, 2005.

[9] D. Applegate and E. Cohen, “Making Intra-Domain Routing Robust to Changing and Uncertain Traffic Demands: Understanding Fundamental Tradeoffs”, in *SIGCOMM ’03*, 2003.

[10] M. Johansson and A. Gunnar, “Data-driven Traffic Engineering: techniques, experiences and challenges”, in *BROADNETS ’06*, 2006.

[11] H. Wang, H. Xie, L. Qiu, Y. Yang, Y. Zhang, and A. Greenberg, “COPE: Traffic Engineering in Dynamic Networks”, in *SIGCOMM ’06*, 2006.

[12] I. Juva, “Robust Load Balancing”, in *GLOBECOM ’07*, 2007.

[13] A. Elwalid, C. Jin, S. Low, and I. Widjaja, “MATE: MPLS Adaptive Traffic Engineering”, in *INFOCOM ’01*, 2001.

[14] S. Kandula, D. Katabi, B. Davie, and A. Charny, “Walking the Tightrope: Responsive yet Stable Traffic Engineering”, in *SIGCOMM ’05*, 2005.

[15] S. Fischer, N. Kammenhuber, and A. Feldmann, “REPLEX: dynamic traffic engineering based on wardrop routing policies”, in *CoNEXT ’06*, 2006.

[16] F. Larroca and J.L. Rougier, “Minimum-Delay Load-Balancing Through Non-Parametric Regression”, in *Networking ’09*, 2009.

[17] P. Casas, L. Fillatre, and S. Vaton, “Robust and Reactive Traffic Engineering for Dynamic Traffic Demands”, in *NGI ’08*, 2008.

[18] L. Kleinrock, “Queueing Systems”, *Wiley-Interscience*, 1975.

[19] Y. Zhang, “Abilene Dataset 04”, http://www.cs.utexas.edu/~yzhang/research/AbileneTM/.

[20] The Abilene Observatory, http://abilene.internet2.edu/observatory/.

[21] K. Cho, “WIDE-TRANSIT 150 Megabit Ethernet Trace 2008-03-18”, http://mawi.wide.ad.jp/mawi/samplepoint-F/20080318/.