HAL
open science

Multi hour robust routing and fast load change detection

Pedro Casas Hernandez, Lionel Fillatre, Sandrine Vaton

To cite this version:

Pedro Casas Hernandez, Lionel Fillatre, Sandrine Vaton. Multi hour robust routing and fast load change detection. ICC 2008 : IEEE International Conference on Communications, May 2008, Beijing, China. pp.5777-5782, 10.1109/ICC.2008.1081 . hal-00540896

HAL Id: hal-00540896

https://hal.science/hal-00540896v1

Submitted on 29 Nov 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi Hour Robust Routing and Fast Load Change Detection for Traffic Engineering

Pedro Casas
TELECOM Bretagne
Brest, France
Email: pedro.casas@telecom-bretagne.eu

Lionel Fillatre
UTT
Troyes, France
Email: lionel.fillatre@utt.fr

Sandrine Vaton
TELECOM Bretagne
Brest, France
Email: sandrine.vaton@telecom-bretagne.eu

***Abstract*—Traffic Engineering (TE) has become a challenging mechanism for network management and resources optimization due to the uncertainty and the difficulty to predict current traffic patterns. Recent works have proposed robust optimization techniques to cope with uncertain traffic, computing a stable routing configuration that is immune to demand variations within certain *uncertainty set*. However, using a single routing configuration for long-time periods can be highly inefficient. Even more, the presence of abnormal and malicious traffic has magnified the network operation problem, claiming for solutions which not only deal with traffic uncertainty but also allow to identify faulty traffic. In this paper, we propose two complementary methods to tackle both problems. Based on expected traffic patterns, we adapt the uncertainty set and build a multi-hour yet robust routing scheme that outperforms the stable approach. For the case of anomalous and unexpected traffic, we propose a fast anomaly detection/isolation algorithm which relies on a novel linear spline-based model of traffic demands to identify traffic problems and decide routing changes. This algorithm is optimal in the sense that it minimizes the decision delay for a given mean false alarm rate and false isolation probabilities. Both proposals are validated using real traffic data from two Internet backbone networks.**

I. Introduction

Traffic engineering (TE) represents a major issue for network operators in today's scenario. TE allows the optimization of network resources usage through multiple mechanisms. In this work, we focus on routing optimization over an Autonomous System (AS). This optimization is becoming increasingly difficult due to the dynamic nature of current traffic. Traffic demands present two different components or *behaviors*: on one hand, a stable and predictable component due to normal traffic usage patterns (e.g. daily demand fluctuation); on the other hand, an abrupt and unpredictable behavior due to unexpected events, such as network equipment failures, flash crowd occurrences, security threats (e.g. denial of service attacks, virus propagation), external routing changes (e.g. inter-AS routing through BGP) and new spontaneous overlay services (e.g. P2P applications). We use the term *volume anomaly* [17] to describe these unexpected network events (large and sudden link load changes).

Recent works [2]–[5] have proposed a new perspective to the routing optimization under traffic uncertainty: the **Robust Routing (RR)** approach. In a robust approach of TE, demand uncertainty is taken into account directly within the routing optimization, computing a single routing configuration for all demands within an *uncertainty set*. While this routing configuration is not optimal for any single traffic matrix (TM) within the set, it minimizes the worst case performance over the whole set. RR provides performance guarantees (i.e. worst-case bounds) for all possible traffic variations within the uncertainty set. However, applying a single robust configuration in the presence of highly variable traffic raises a difficult question: how should this uncertainty set be defined? Larger sets cover a broader group of possible demands, but at the cost of routing inefficiency. On the other hand, tighter sets produce more efficient routing schemes, but subject to poor performance guarantees. RR presents another drawback: it does not solve the problem of faulty traffic identification. The early detection and isolation of unusual and significant changes in traffic demands allows not only to perform an accurate routing reconfiguration but also provides additional information for improving network operation.

A. Related Work

There is a large literature on traffic engineering with uncertain traffic demands. Traditional algorithms rely on a small group of expected TMs (representative traffic demands from past observations) or estimated TMs to compute optimal and reliable routing configurations. An extreme case is presented in [12], where routing is optimized for a single estimated TM and it is then applied for long-time periods (24hs periods). Traffic uncertainty is characterized by multiple TMs in [13], [14] (e.g. set of TMs from previous day, same day of previous week, etc.), and different ways to find optimal routes for the set are presented. Given the dynamic nature of present demands, this perspective is no longer suitable for current scenario [1]. A different approach is provided by online reactive algorithms: TeXCP [15] and MATE [16] both balance load in realtime, responding to instantaneous traffic demands. Their main goal is to avoid network congestion by adaptively balancing the load among paths, based on measurement. Reactive routing presents a desirable property, that of keeping routing adapted to current traffic. However, these adaptive algorithms present poor performance under significant and abrupt traffic changes [5]. A third category of algorithms consists in Stable Robust Routing techniques [2]–[6]. In [2], the authors capture traffic variations by introducing a polyhedral set of demands, ap-

plying linear programming techniques to compute an optimal stable routing for all demands within this set. [4] applies this robust technique to compute a robust MPLS routing configuration without depending on TM estimation. Oblivious Routing [3] also defines linear algorithms to optimize worst-case performance for different sizes of traffic uncertainty sets, aiming to handle dynamic changes. [6] analyses the use of robust routing through a combination of traffic matrix estimation and its corresponding estimation error bounds, in order to shrink the uncertainty set. The drawback of stable robust routing is its inherent dependence on the definition of the uncertainty set: 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.

As regards anomaly detection in data networks, the problem has been extensively studied. In this section, we will just overview those works that have motivated our signal-processing based detection algorithm. Signal processing techniques have been applied to the anomaly detection field [9]–[11]. The usual behavior of data flows is modeled by several approaches: spectral analysis, time series analysis, wavelets decomposition, etc. Anomalies correspond to deviations from the usual behavior of the data flows. The general flaw of these algorithms is the lack of stability over time of the proposed traffic models, as well as the absence of optimality conditions for the detection in most cases. A second class of methods related to our model concerns statistical hypotheses testing [17]–[19]. When data flows are parametrically modeled, the design of optimal algorithms is possible. Nevertheless, non-parametric approaches are particularly studied because of the lack of parametric models, and these approaches are often sub-optimal. The detection/isolation of traffic anomalies problem was previously treated in [17], using a TM decomposition on the Principal Component Analysis (PCA) basis. However, this approach presents a major stability problem: the PCA basis depends on the measurement period, rendering it unstable over time.

B. Contributions of the Paper

We propose two novel and complementary approaches to deal with current dynamic traffic demands, separately treating both traffic uncertainty sources. For ***expected traffic fluctuations***, we present a time varying approach of RR that outperforms the current *stable* approach: the **Multi-Hour Robust Routing (MHRR)**. We preserve the virtues of RR, but change the routing configuration during time. The uncertainty set is optimally divided into several uncertainty sub-sets that better adapt to real traffic loads and a stable robust routing scheme is computed for each sub-set. The partitioning algorithm allows to calculate the exact moments when routing changes must be performed. For the case of ***unpredictable traffic behavior***, we propose a novel volume anomaly detection/isolation algorithm to identify traffic problems and decide routing changes. To overcome the limitations of the PCA approach [17], we propose a non data-driven traffic model which remains stable over time. This model allows to separate normal from anomalous traffic, based on simple link load measurements. Both proposals are validated using real traffic data from two backbone networks, the Internet2 Abilene backbone network and a private international Tier-2 network.

The remainder of this paper is organized as follows. In Section II, we recall the basic aspects of the robust routing approach. Section III presents the theoretical background and empirical evaluation of the MHRR. The proposed algorithm and traffic model for anomaly detection/isolation are introduced and validated in section IV. Finally, Section V concludes this work.

II. Robust Routing

Let us consider a network topology defined by a set of n nodes and $L = \{1, \ldots, r\}$ links with capacities in $C = (c_1, c_2, \ldots, c_r)$. The TM demand $\mathbf{d} = \{d_{i,j}\}$ denotes the traffic flow between every node i and node j ($i \neq j$) of the network. We re-arrange $\mathbf{d}$ as a column vector, $\mathbf{d} = \{d_{k,\ k=1..m}\}$, where d_k represents the traffic flow transmitted by OD pair k (OD-flow k) and $m = n \times (n-1)$ is the number of OD pairs. Let $N = \{\text{OD}_1, \ldots, \text{OD}_m\}$ be the set of OD pairs. Link's information y_l represents the total traffic (i.e. aggregated OD flows) through link l in a certain period of time. This information is available from router's MIB variables and it is usually collected every 5' periods via SNMP [20]. Traffic demands and links' traffic are related through the routing matrix R, a $r \times m$ matrix $R = \{r_{l,k}\}$ where $0 \leqslant r_{l,k} \leqslant 1$ represents the fraction of OD demand k routed through link l:

$$\mathbf{y} = R \times \mathbf{d}. \tag{1}$$

with $\mathbf{y} = \{y_{l,\ l=1..r}\}$. Routing optimization depends on the underlying data transport mechanism; we will focus on path-based routing such as MPLS. This optimization consists of minimizing certain performance metric associated with traffic demand. Throughout this work we consider maximum link utilization (MLU) as the routing performance criterion. Overloaded links tend to cause QoS degradation (e.g. larger delays and packet losses, throughput reduction, etc.), so MLU represents a reasonable measure of network performance. The are many other performance metrics that could be used instead of MLU, like path's end-to-end delay or mean link utilization; setting the focus too strictly on the MLU can often lead to longer average traffic paths and thus adversely affect the mean. However, we use the MLU as it is the most commonly applied criterion and it represents an easy to understand performance metric. For a given routing matrix $R = \{r_{l,k}\}$ and a traffic demand $\mathbf{d}$, the MLU (u_{max}) is defined as the maximum of the ratio between link load and link capacity:

$$u_{max}\left(C, \mathbf{d}, R\right) = \max_{l \in \{1 \ldots r\}} \sum_k \frac{r_{l,k} \cdot d_k}{c_l} = \max_{l \in \{1 \ldots r\}} \frac{y_l}{c_l} \tag{2}$$

Let $P(k)$ be the set of possible paths for OD demand k. Let x_p^k be the proportion of traffic demand d_k that flows through path $p \in P(k)$, $0 \leqslant x_p^k \leqslant 1$. Finally, let x_l^k be the proportion of traffic demand d_k that flows through link $l \in L$, $0 \leqslant x_l^k \leqslant 1$. We define D as the uncertainty set

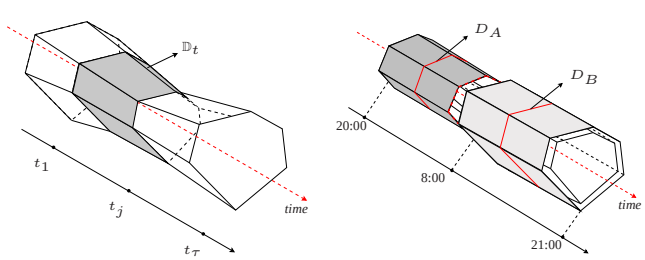
where traffic demand may vary. This set can be defined in different ways, depending on the available information: link load measurements and historical routing, a set of previously observed TMs $\{\mathbf{d}^1, \mathbf{d}^2, \ldots, \mathbf{d}^o\}$, TM time series $\mathbf{d}(t)$, etc. [2] defines this set as a *polytope*, based on the intersection of several half-spaces that result from linear constraints imposed to traffic demand. The Robust Routing Optimization Problem (RROP) consists of minimizing u_{max}, considering all demands within D (3). The RROP can be efficiently solved by linear

$$\begin{array}{rll} \text{minimize} & u_{max} & \\ \text{subject to:} & & \\ \sum\limits_{p \in P(k)} x_p^k & \geqslant 1 & \forall\, k \in N \\ \sum\limits_{p \in P(k),\ l \in p} x_p^k & \leqslant x_l^k & \forall\, k \in N,\ \forall\, l \in L \\ \sum\limits_{k \in N} x_l^k . d_k & \leqslant u_{max} \cdot c_l & \forall\, l \in L,\ \forall\, \mathbf{d} \in D \\ x_p^k,\ x_l^k & \geqslant 0 & \forall\, l \in L, \forall\, p \in P(k),\ \forall\, k \in N \\ u_{max} & \leqslant 1 & \end{array} \tag{3}$$

programming techniques, applying a combined column and constraint generation method [2]. In a traditional robust routing application, the obtained routing configuration is applied during long-term periods of time (i.e. daily routing). In this sense, we refer to robust routing as **Stable Robust Routing (SRR)**.

III. Multi-Hour Robust Routing

In [1] we present the advantages of the SRR with respect to traditional routing approaches: SRR offers stability guarantees against traffic uncertainty and traffic time-variations at a reasonable cost. However, considering a single routing scheme for long-time periods is conservative and results in sub-optimal performance. We propose a simple approach to shrink and adapt the uncertainty set along time that outperforms the SRR. Based on rough knowledge of traffic variations (i.e. considering expected traffic behavior), we propose to optimally divide the uncertainty set and build a multi-hour routing configuration, considering a single SRR configuration for each sub-set. Daily traffic changes can be seen as a time variation of



(a) (b)

Fig. 1. (a) Daily variation of the polytope D_t, (b) time partitioning of D_t.

the uncertainty set. At each time t, the routing matrix R and the link load values $\mathbf{y}(t) = \mathbf{y}^t$ define an instantaneous uncertainty set $D(t) = \{\mathbf{d} \in \mathbb{R}^m, R \times \mathbf{d} \leqslant \mathbf{y}^t,\ \mathbf{d} \geqslant 0\}$. The continuous union of infinite instantaneous uncertainty sets along time t defines the *daily uncertainty set* $\mathbb{D}_t = \left\{(\mathbf{d}, t) \in \mathbb{R}^{m+1},\ \mathbf{d} \in \cup_{t_1 \leqslant t \leqslant t_\tau} D(t),\ t_1 \leqslant t \leqslant t_\tau\right\}$. Figure 1(a) explains this idea. Assuming this set is an union of polytopes, [7] provides a theoretical study of the optimal partitioning of $\mathbb{D}_t$, using a partitioning hyper plane. [7] proves that this is a NP-hard problem, except for the case where a partitioning direction is previously fixed. We define a partitioning hyper plane by its direction vector α and a value w: $\alpha.\mathbf{d} = w$. In the MHRR approach, we consider a particular direction for partitioning: the *time direction*. In that case, w represents the time of the day. We define $h+1$ hyperplanes at times $\{w_1, w_2, .., w_{h+1}\}$. The intersection between $\mathbb{D}_t$ and the half-spaces defined by these partitioning hyperplanes results in h uncertainty sub-sets $\mathbb{D}_i = \{\mathbb{D}_t \cap \{\mathbf{d}, \alpha.\mathbf{d} \geqslant w_i\} \cap \{\mathbf{d}, \alpha.\mathbf{d} \leqslant w_{i+1}\}\}, \forall i = 1, ., h$. Let D_i be the smallest single-time set that contains all demands $\mathbf{d}(t) \in \mathbb{D}_i, w_i \leqslant t \leqslant w_{i+1}$ (see figure 1(b)). A SRR configuration R^i_{robust} is computed for each sub-set D_i. Each routing configuration is finally applied at each time interval. The optimal values of routing changes $\mathbf{w}^* = \{w_2^*, \ldots, w_h^*\}$ are the solution for the following optimization problem (w_1 and w_{h+1} are fixed a priori, as they define the considered time interval of analysis):

$$\mathbf{w}^*(\mathbb{D}_t) = \arg\min_{\mathbf{w}} \left\{ \max_{i=1..h} u_{max}(D_i) \right\} \tag{4}$$

where $u_{max}(D_i)$ is the solution for (3) for polytope D_i. [7] presents a simple algorithm to approximately solve (4) (within an arbitrary precision), using a generalization of a simple dichotomy methodology. The MHRR presents a trade-off between performance and routing stability. The more intervals we use, the more adapted the routing becomes. However, the number of intervals should be bounded as many routing changes may lead to instabilities and performance degradation. In a general case, 2 sub-sets are enough to handle the usual daily variation.

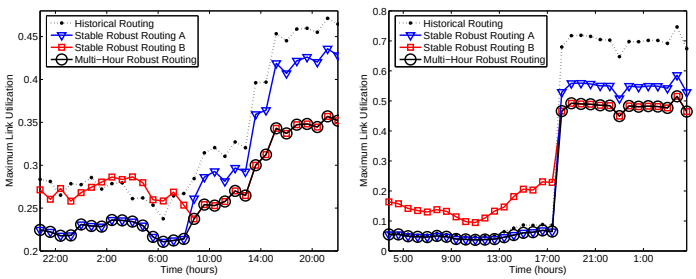
MHRR Evaluation

We present a comparative analysis between SRR and MHRR in Abilene, an Internet2 backbone network. Abilene consists of 12 router-level nodes and 30 OC192 links (2 OC48). The used router-level network topology and traffic demands are available at [25]. Traffic data consists in 6-month traffic matrices collected every 5' via Netflow from the Abilene Observatory [26]. The time-variation of the polytope is not a simple homothety [1]; in this sense, we will show that a routing configuration change during the day improves routing performance. Let R_o be the historical routing matrix of Abilene, not necessarily optimal (R_o is available at [25]). We consider a single time partitioning (i.e. 2 routing intervals), $w_1 = 20{:}00$, $w_2 = w^*$ and $w_3 = 21{:}00$, where w^* is the solution for (4). For each time interval, we consider the smallest polytope that includes all possible realizations over that period:

$$D_{A,B} = \{\mathbf{d} \in \mathbb{R}^m, R_o \times \mathbf{d} \leqslant \mathbf{y}_{A,B},\ \mathbf{d} \geqslant 0\} \tag{5}$$

where $\mathbf{y}_A = \mathbf{y}_{max}^{20:00-w^*}$ and $\mathbf{y}_B = \mathbf{y}_{max}^{w^*-21:00}$ (maximum values for each link). In this way, D_A includes all traffic

demands between 20:00 and w^* and D_B between w^* and 21:00 (see figure 1(b)). For each polytope, we compute a SRR configuration, R^A_{robust} and R^B_{robust}. In order to compare stable and multi-hour approaches, we apply both routing configurations during the whole evaluation period. We include the routing performance obtained with R_o (curve *historical routing*) to appreciate the time variation of traffic loads. Figure



(a) Expected daily behavior (b) Anomalous unexpected event

Fig. 2. Routing performance, stable vs. multi-hour robust routing.

2(a) compares the routing performance (MLU) between these two RR configurations. Polytope D_A is well suited for smaller loads, so R^A_{robust} performs better during the first half of the day, when network load is lower. However, when traffic increases, demands that do not belong to D_A produce higher link utilizations than those obtained with R^B_{robust}. The MHRR consists in computing the time when routing must be changed ($w^* \approx$ 8:00 in this case), using the corresponding routing configuration depending on the time of the day (R^A_{robust} before w^* and R^B_{robust} after). The MHRR approach presents a performance improvement of 15% with respect to the SRR approach before w^*, reaching a near 20% of over-efficiency after w^*. We repeat the same evaluation but considering a traffic demand that drastically changes (i.e. a large time-variation of the polytope, caused by a volume anomaly). Figure 2(b) presents an abrupt change in MLU (almost 14 times higher) at time 18:00. In this case, we assume that this change is known in advance (note that in the general case, it is not possible to predict these abrupt changes). The optimal moment for changing routing is $w^* \approx$ 18:00. The MHRR approach definitely outperforms the SRR in this experience, presenting a MLU between 10% and 60% smaller during the whole evaluation period.

IV. Dealing with Unexpected Events

The proposed MHRR approach offers a robust and efficient routing configuration, given a rough knowledge of the daily uncertainty set. However, in the presence of volume anomalies it is no longer possible to apply the MHRR as the daily uncertainty set is unknown. For those cases, we propose a fast volume anomaly detection/isolation algorithm to quickly identify faulty traffic. This detection allows to decide as soon as possible the moment when routing configuration must be changed. The goal of the algorithm is to detect/isolate an additive change $\boldsymbol{\theta}$ in the time series of traffic demand $\mathbf{d}(t)$ from a sequence of link load measurements $\mathbf{y}(t) = R \times \mathbf{d}(t)$. We use link loads as input to avoid relying on seldom available traffic demands. In this work, we focus on detecting and isolating a "localized" anomaly, $\boldsymbol{\theta} = \theta\left(\delta_{1,i}, \ldots, \delta_{i,i}, \ldots, \delta_{m,i}\right)^T$, where $\delta_{i,j} = 0$ if $i \neq j$ and $\delta_{i,i} = 1$ (this corresponds to a change θ in OD flow i). If several OD flows are simultaneously corrupted, the detection/isolation algorithm produces an alarm and identifies only one faulty OD flow. The algorithm can be extended to detect/isolate simultaneous anomalies, but its' complexity (n^o hypotheses, see IV-B) grows highly. The isolation of the anomalous traffic is possible since an anomaly in a given OD flow typically spans multiple links. Real traffic demands follow a *non-observable* model from link load measurements: since $r < m$, it is impossible to retrieve $\mathbf{d}(t)$ from $\mathbf{y}(t)$ without additional assumptions on the traffic demand. To overcome this difficulty, we propose a parsimonious linear model for non-anomalous traffic. This model renders traffic demands observable and therefore, it allows to separate usual from anomalous traffic.

A. Stochastic Traffic Model for Anomaly Detection

We assume that the stochastic process of the OD traffic demand $\mathbf{d}(t)$ obeys the following linear model:

$$\mathbf{d}(t) = \boldsymbol{\lambda}(t) + \boldsymbol{\xi}(t) \tag{6}$$

where $\boldsymbol{\lambda}(t) \in \mathbb{R}^m$ is the mean traffic demand and $\boldsymbol{\xi}(t)$ is a white Gaussian noise with covariance matrix $\Sigma(t)$ that represents the model error together with the natural variability of the OD flows (based on the results obtained in [22]). The process $\boldsymbol{\lambda}(t)$ represents the "regular" part of the OD TM which can be correctly modeled when the behavior of the network is anomaly-free. We propose to parameterize this vector by exploiting the stationarity of the spatial distribution of the TM. One of the few invariants of Internet traffic is that a small

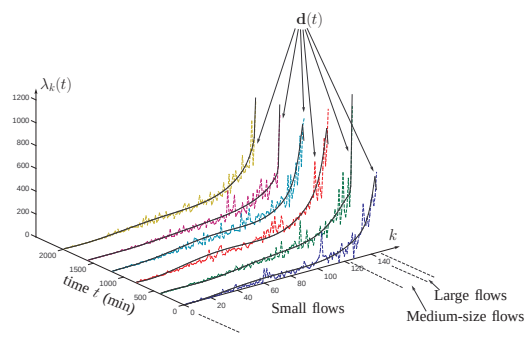


Fig. 3. Approximation of real OD flows by the spline-based model.

percentage of flows contribute to a large proportion of total traffic [4], [8]. Hence, if we assume that the traffic distribution between the different OD couples is spatially stationary in the absence of an anomaly, the order of increasing OD flows remains constant during long time periods. The proposed traffic model takes advantage of the stationary property of this ordering. We propose to classify OD flows in three different classes, depending on their volume: large OD flows, small OD flows and medium-size OD flows. The sorted components

can be interpreted as a discrete increasing signal. The curve obtained by interpolating this discrete signal is assumed to be a continuous curve, hence it can be parameterized by using a polynomial approximation. Figure 3 shows the OD flows, sorted in the increasing order of their volume of traffic, as a function of the time t. Since data are vectors of finite dimension, we propose to use the following method to design a discrete spline basis: (i) we choose a continuous spline basis; (ii) we discretize all these splines according to m points uniformly chosen in the interval $[1; m]$ and (iii) we rearrange all these discrete signals according to previous sorting order. We finally obtain the following linear model for the anomaly-free traffic demand:

$$\mathbf{d}(t) = S\boldsymbol{\mu}(t) + \boldsymbol{\xi}(t) \tag{7}$$

where $S = (\mathbf{s}_1\, \mathbf{s}_2 \ldots \mathbf{s}_q)$ is a $m \times q$ known matrix with columns $\mathbf{s}_j$ and q is small with respect to m. The vectors $\mathbf{s}_i$, which correspond to the rearranged discrete spline, form a set of known basis vectors describing the spatial distribution of the traffic and $\boldsymbol{\mu}(t) = (\mu_1(t) \ldots \mu_q(t))^T$ is the unknown time varying parameter vector which describes the OD flow intensity distribution with respect to the set of vectors $\mathbf{s}_i$. The model for the anomaly-free link traffic is given by:

$$\mathbf{y}(t) = H\boldsymbol{\mu}(t) + \boldsymbol{\zeta}(t), \tag{8}$$

where $H = RS$ and $\boldsymbol{\zeta}(t) = R\boldsymbol{\xi}(t)$. In this way, we can describe the usual behavior of traffic demands from simple link measurements. The computation of the rank of H is not simple since it depends on the routing matrix R. In practice, since the number of columns of H is very small, the product RS and its rank can be computed very fast. Therefore, we will assume that H is full column rank. Finally, the covariance matrix Σ is unknown. The remedy consists in computing an estimate $\widehat{\Sigma}$ of Σ. Results on the estimation of $\widehat{\Sigma}$ can be found in [21].

B. Volume Anomaly Detection/Isolation

The detection/isolation of a volume anomaly at time t_0 can be treated as a hypothesis testing problem where the null hypothesis $\mathcal{H}^0_{t_0} = \{$the OD flows are anomaly-free at time $t_0\}$ is tested against m alternatives $\mathcal{H}^j_{t_0} = \{$the j-th OD flow presents an anomalous additional amount of traffic θ from the time $t_0\}$. The change detection algorithm has to compute a pair (T, ν), where T is the alarm time at which a ν-type change ($\nu \in \{1, 2, \ldots, m\}$) is detected and isolated, based on link traffic observations $\mathbf{y}_1, \mathbf{y}_2, \ldots$ The hypothesis testing can be written as

$$\mathcal{H}_0 \;:\; \mathbf{y}(t) \sim \mathcal{N}(H\,\boldsymbol{\mu}(t), R\Sigma R^T)\,, \quad t = 1, 2, \ldots, \tag{9}$$

$$\mathcal{H}^j_{t_0} \;:\; \begin{cases} \mathbf{y}(t) \sim \mathcal{N}(H\,\boldsymbol{\mu}(t), R\Sigma R^T), \; t = 1, \ldots, t_0 - 1, \\ \mathbf{y}(t) \sim \mathcal{N}(H\,\boldsymbol{\mu}(t) + \theta_j\,\mathbf{r}_j, R\Sigma R^T)\,, \\ \qquad \theta_{j,1} \leqslant |\theta_j| \leqslant \theta_{j,2}\,, \; t = t_0, t_0 + 1, \ldots \end{cases} \tag{10}$$

where $\mathbf{r}_{j,\,1 \leqslant j \leqslant m}$ denotes the normalized j-th column of R and $0 < \theta_{j,1} < \theta_{j,2} < +\infty$ are some known bounds on the change intensity of the j-th OD flow (these bounds are introduced for technical reasons but they can be chosen arbitrarily). As we show in the Appendix, we can simplify this problem by eliminating the non-anomalous traffic. In this case, hypothesis (10) can be rewritten as

$$\mathcal{H}^j_{t_0} : \begin{cases} \mathbf{z}(t) \sim \mathcal{N}(0, I_{r-q})\,, \quad t = 1, \ldots, t_0 - 1, \\ \mathbf{z}(t) \sim \mathcal{N}(\theta_j \mathbf{v}_j, I_{r-q})\,, \\ \qquad \theta_{j,0} \leqslant |\theta_j| \leqslant \theta_{j,1}\,, \quad t = t_0, t_0 + 1, \ldots \end{cases} \tag{11}$$

where $\mathbf{v}_j$ is a known vector and $\mathbf{z}(t)$ are the normalized residuals obtained from $\mathbf{y}(t)$ after filtering the non-anomalous traffic. The vector $\mathbf{v}_j$ corresponds to the signature in the residuals of a change in OD flow j. We use the optimal recursive algorithm (T_r, ν_r) proposed in [24] to solve (11) :

$$T_r \;=\; \min_{1 \leqslant k \leqslant m} \{T_r(k)\}, \quad \nu_r \;=\; \arg \min_{1 \leqslant k \leqslant m} \{T_r(k)\}$$

$$T_r(k) = \inf \left\{ t \geqslant 1 : \min_{0 \leqslant j \neq k \leqslant m} [g_t(k, j) - h_{k,j}] \geqslant 0 \right\} \tag{12}$$

with $g_t(k, j) = g_t(k, 0) - g_t(j, 0)$. The recursive functions $g_t(k, 0)$ are defined by

$$g_t(k, 0) \;=\; (g_{t-1}(k, 0) + z_t(k, 0))^+ \tag{13}$$

$$z_t(k, 0) \;=\; \log \frac{f_k(\mathbf{z}(t))}{f_0(\mathbf{z}(t))} \tag{14}$$

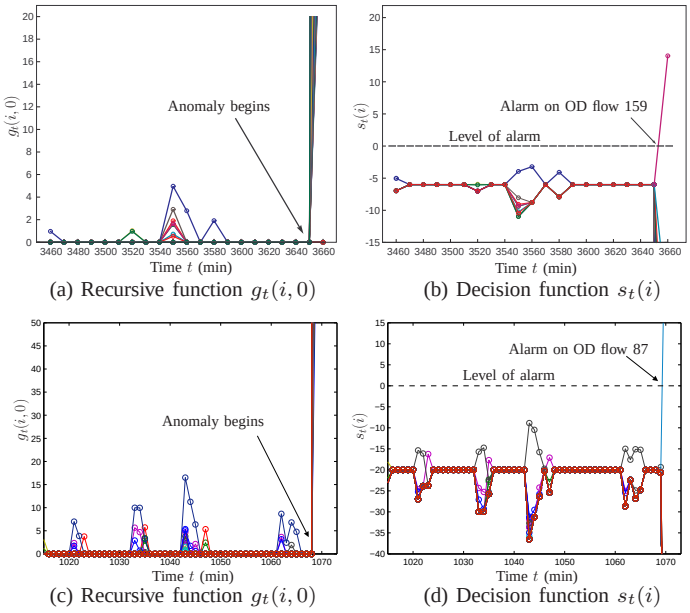
$g_0(k, 0) = 0$ for every $1 \leqslant k \leqslant m$ and $g_t(0, 0) = 0$ for all t. f_0 represents the probability density function of anomaly-free traffic measurements. f_k is the probability density function of residuals $\mathbf{z}(t_0), \mathbf{z}(t_0 + 1), ..$ after a change of type k. The thresholds $h_{k,j}$ are chosen by the following formula:

$$h_{k,j} = \begin{cases} h_{\mathrm{d}} & \text{if } 1 \leqslant k \leqslant m \quad \text{and } j = 0 \\ h_{\mathrm{i}} & \text{if } 1 \leqslant k, j \leqslant m \quad \text{and } j \neq k \end{cases}$$

where h_{d} is the detection threshold and h_{i} is the isolation threshold. For given bounds γ and β, this algorithm is asymptotically optimal, i.e. it reaches the lower bound of the maximum mean delay for detection [24]. The choice of the detection and isolation thresholds h_{d} and h_{i} is discussed (with practical comments and simulations) in [23].

C. Validation

We demonstrate the ability of the detection/isolation algorithm to detect and identify a volume anomaly in SNMP link flow data from two different networks (different not only in the topology but also in the behavior of traffic demands): a large Tier-2 network (50 nodes, 168 measured links and 2450 non-zero OD flows, sampled at a 10 minute rate) and Abilene (the Abilene dataset consists in Netflow traces, so we use the supplied routing matrix to retrieve link loads). Figure 4 shows the typical realizations of the decision functions $g_t(i, 0)$ and $s_t(i) = \min_{0 \leqslant i \neq k \leqslant m} [g_t(i, k) - h_{i,k}]$ vs the elapsed time. The functions $s_t(i)$ are used to "monitor" the OD flows; when the function $s_t(i)$ exceeds 0, OD flow i is declared faulty. It is assumed that the anomaly in the Tier-2 network begins at time 3660, and at time 1070 in Abilene. Note that after this time, several decision functions $g_t(i, 0)$ rapidly grow. Each function $g_t(i, 0)$ is associated with OD flow i and when this function grows, it means that OD flow i is suspected of carrying an abnormal amount of traffic. Contrary to $g_t(i, 0)$, only decision function $s_t(159)$ ($s_t(87)$ in Abilene) associated to faulty OD flow 159 (87 respectively) grows and finally exceeds the



(a) Recursive function $g_t(i,0)$ (b) Decision function $s_t(i)$

(c) Recursive function $g_t(i,0)$ (d) Decision function $s_t(i)$

Fig. 4. Typical realizations of decision functions for a Tier-2 network (a,b) and Abilene (c,d).

threshold. Hence, the functions $s_t(i)$ permit us to isolate the faulty OD flow among all the OD flows associated to functions $g_t(i,0)$ that have rapidly grown. At time 3660 (1070 respectively), an alarm is raised and the algorithm selects the faulty OD flow 159 (87 respectively). The decision function $s_t(i)$ needs only 1 observation (10 minutes in the Tier-2 network or 5 minutes in Abilene, but this is the smallest delay than can be achieved given these sampling-rates) to detect and isolate the faulty OD flow. An interesting observation is that the detection/isolation algorithm achieves good results in both networks, even though the respective traffic demand behaviors are completely different between these two networks.

V. Conclusions and Future Work

In this paper, we address the routing under traffic uncertainty problem. We provide a solution that not only deals with current dynamic traffic demands in a robust and efficient way but also detects and isolates large-volume anomalous traffic, improving network operation. We extend the robust routing paradigm by introducing the notion of time-varying uncertainty set, setting up a multi-hour robust routing scheme. We show that this approach achieves better resource utilization than previous stable robust proposals in different scenarios. We introduce an original linear spline-based parsimonious model to parameterize normal traffic behavior from widely available link load measurements. Based on this model, we present a statistical algorithm to detect and isolate volume anomalies in network traffic. We apply this algorithm to cope with sudden and large traffic changes in current dynamic demands, complementing the multi-hour robust scheme. Many important issues remain open for further study. A deep evaluation of the challenges involved in changing routing configuration after the detection of the anomalous traffic should be performed. The isolation ability of the proposed algorithm can be highly exploited in order to recompute routing in a more efficient way. The impact of routing re-configuration on end-to-end traffic must be explored, especially considering the imposed QoS restrictions in the current end-user Internet-services scenario.

References

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

[2] W. Ben-Ameur and H. Kerivin, "Routing of Uncertain Traffic Demands", in *Optimization and Engineering*, 6, 283-313, 2005.

[3] D. Applegate and E. Cohen, "Making Intra-Domain Routing Robust to Changing and Uncertain Traffic Demands: Understanding Fundamental Tradeoffs", in *Proc. ACM SIGCOMM*, 2003.

[4] M. Johansson and A. Gunnar, "Data-driven Traffic Engineering: techniques, experiences and challenges", in *Proc. BROADNETS 2006*, 2006.

[5] H. Wang, H. Xie, L. Qiu, Y. Yang, Y. Zhang and A. Greenberg, "COPE: Traffic Engineering in Dynamic Networks", in *Proc. ACM SIGCOMM*, 2006.

[6] I. Juva, "Robust Load Balancing", in *Proc. IEEE GLOBECOM*, 2007.

[7] W. Ben-Ameur, "Between Fully Dynamic Routing and Robust Stable Routing", in *Proc. DRCN 2007*, 2007.

[8] A. Medina, K. Salamatian, S. Bhattacharyya and C. Diot, "Traffic Matrix Estimation: Existing Techniques and New Directions", in *Proc. ACM SIGCOMM*, 2002.

[9] P. Barford, J. Kline, D. Plonka, and A. Ron, "A Signal Analysis of Network Traffic Anomalies", in *ACM SIGCOMM Internet Measurement Workshop*, 2002.

[10] Brutlag, Jake D., "Aberrant Behavior Detection in Time Series for Network Monitoring", in *Proc. 14th Systems Administration Conference*, 2000.

[11] C.M. Cheng, H. Kung, K.S. Tan, "Use of Spectral Analysis in Defense Against DoS Attacks", in *Proc. IEEE GLOBECOM*, 2002.

[12] M. Roughan, M. Thorup and Y. Zhang, "Traffic Engineering with Estimated Traffic Matrices", in *Proc. USENIX/ACM Internet Measurement Conference*, 2003.

[13] C. Zhang, Z. Ge, J. Kurose, Y. Liu and D. Towsley, "Optimal Routing with Multiple Traffic Matrices: Tradeoff between Average case and Worst case Performance", in *Proc. 13th International Conference on Network Protocols (ICNP)*, 2005.

[14] C. Zhang, Y. Liu, W. Gong, J. Kurose, R. Moll and D. Towsley, "On Optimal Routing with Multiple Traffic Matrices", in *Proc. IEEE INFOCOM*, 2005.

[15] S. Kandula, D. Katabi, B. Davie and A. Charny, "Walking the Tightrope: Responsive yet Stable Traffic Engineering", in *Proc. ACM SIGCOMM*, 2005.

[16] A. Elwalid, C. Jin, S. Low and I. Widjaja, "MATE: MPLS Adaptive Traffic Engineering", in *Proc. IEEE INFOCOM*, 2001.

[17] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing Network-Wide Traffic Anomalies", in *Proc. SIGCOMM*, 2004.

[18] A. Lakhina et all, "Characterization of Network-Wide anomalies in Traffic Flows", in *IMC*, 2004.

[19] M. Thottan and C. Ji, "Anomaly Detection in IP Networks", in *IEEE Trans. on Signal Processing*, 2003.

[20] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford and F. True, "Deriving Traffic Demands for Operational IP Networks", in *IEEE/ACM Trans. on Networking*, vol. 9, no. 3, pp. 265-279, 2001.

[21] Y. Vardi, "Network Tomography: Estimating Source-Destination Traffic Intensities from Link Data", in *Journal of American Statistical Association*, 1996.

[22] M. Coates, A. Hero, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Processing Mag.*, May 2002.

[23] I. Nikiforov, "A simple recursive algorithm for diagnosis of abrupt changes in random signals", in *IEEE Trans. on IT*, vol. 46, no. 7, pp. 2740-2746, 2000.

[24] I. Nikiforov, "A lower bound for the detection/isolation delay in a class of sequential tests", in *IEEE Trans. on IT*, vol. 49, no. 11, pp. 3037-3046, 2003.

[25] Y. Zhang, "Abilene Dataset 04", *http://www.cs.utexas.edu/yzhang/research/AbileneTM/*.

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

Appendix - Elimination of non-anomalous traffic

Non-anomalous traffic $H\boldsymbol{\mu}(t)$ is eliminated by projecting the measurement vector $\mathbf{y}(t)$ on the null space of H. By using the invariant properties of the Gaussian law, the general covariance matrix in (10) is reduced to the identity one. Let us define the matrix $W = (\mathbf{w}_1, .., \mathbf{w}_{r-q})$ of size $r \times (r-q)$ composed of eigenvectors $\mathbf{w}_1, .., \mathbf{w}_{r-q}$ of the projection matrix $P_H^{\perp} = I_r - H(H^T H)^{-1} H^T$ corresponding to eigenvalue 1. The matrix W satisfies the following conditions: $W^T H = 0$, $WW^T = P_H^{\perp}$ and $W^T W = I_{r-q}$. The matrix W can be considered as a linear rejector that eliminates the non-anomalous traffic. Under hypothesis $\mathcal{H}_{t_0}^j$, the sequence $W^T \mathbf{y}(t)$ can be modeled as $W^T \mathbf{y}(t) = W^T \boldsymbol{\zeta}(t) + \theta_j W^T \mathbf{r}_j, \quad j = 1, .., m$. Since $W^T \boldsymbol{\zeta}(t)$ is a correlated Gaussian vector with covariance matrix $\widetilde{\Sigma} = W^T R \Sigma R^T W$, each vector $W^T \mathbf{y}(t)$ is normalized by using the square root matrix $\widetilde{\Sigma}^{\frac{1}{2}}$, $\mathbf{z}(t) = \widetilde{\Sigma}^{-\frac{1}{2}} W^T \mathbf{y}(t) \sim \mathcal{N}(\theta_j \mathbf{v}_j, I_{r-q})$, with $\mathbf{v}_j = \widetilde{\Sigma}^{-\frac{1}{2}} W^T \mathbf{r}_j$.