Automated Detection of Load Changes in Large-Scale Networks

Felipe Mata, Javier Aracil, and Jose Luis García-Dorado

Universidad Autónoma de Madrid, Spain
{felipe.mata,javier.aracil,jl.garcia}@uam.es

Abstract. This paper presents a new online algorithm for automated detection of load changes, which provides statistical evidence of stationary changes in traffic load. To this end, we perform continuous measurements of the link load, then look for clusters in the dataset and finally apply the Behrens-Fisher hypothesis testing methodology. The algorithm serves to identify which links deviate from the typical load behavior. The rest of the links are considered normal and no intervention of the network manager is required. Due to the automated selection of abnormal links, the Operations Expenditure (OPEX) is reduced. The algorithm has been applied to a set of links in the Spanish National Research and Education Network (RedIRIS) showing good results.

Keywords: Load change, capacity planning, Behrens-Fisher problem.

1 Introduction and Problem Statement

The steady growth of Internet traffic [1,2,3] makes it necessary to pay close attention to load changes. Actually, network operators face bandwidth outages, and there is a growing pressure, both from customers and regulatory bodies, to ensure Quality of Service (QoS). Furthermore, operators are currently offering Service Level Agreements in their product portfolios, and the levels of QoS in terms of delay, bandwidth and jitter are very challenging to achieve in practice. Thus, there is an increasing need to detect changes in traffic load in order to perform an adequate capacity planning.

This paper focuses on detection of traffic changes in large-scale networks, i.e. with a very large number of links. In such networks, there are many traffic probes that produce time-series of link occupation (traffic volume). Being the number of links very large, it is not feasible to inspect all the time-series visually, and then make capacity planning decisions. The techniques provided in this paper allow the network manager to focus on those links that show a significant deviation from their typical behavior, and thus call for an upgrade.

On the other hand, we focus on the capacity planning timescale. The proposed detection techniques are amenable to use in the timescale of days or weeks. This is the timescale for capacity planning decisions ([4]), i.e. the timescale to decide whether more bandwidth should be rolled out and in which links. Therefore, this paper does not investigate the issue of reactive response in terms of severe traffic

M. Papadopouli, P. Owezarski, and A. Pras (Eds.): TMA 2009, LNCS 5537, pp. 34–41, 2009.
© Springer-Verlag Berlin Heidelberg 2009

load peaks, which typically happens in the timescale of minutes or below. For this kind of traffic load detection a threshold-based algorithm applies better. On the contrary we focus on links with low-medium load that is increasing continuously over time. More specifically, we look for changes in traffic volume, which require intervention from the network manager, and possibly lead to a capacity planning decision.

The proposed technique employs a combination of clustering algorithms and the Behrens-Fisher test of hypothesis. The main advantage is that it reduces OPEX. Indeed, our technique marks the links as either remaining stable or changing in load and only the latter require human intervention. As a result, the load monitoring tasks are less time-consuming for the network manager.

Concerning the state of the art we find methods for traffic forecasting, such as the one presented in [5]. Our work provides a technique to decide if a link is deviating from its typical behavior but we do not perform traffic forecasting. The authors in [6] propose a model to decide when and to which capacity out of a discrete set is more convenient to upgrade a network link. The model takes into account economic variables such as the revenue, the risk free interest rate and the market price of risk to determine the value of the investment and based on these results the authors decide when is profitable to upgrade. Our work differs from this one because we detect the changes in load instead of running a model to check the network investment periodically. Our approach also diverges from the usual capacity planning studies where a link is marked as a candidate for upgrading when it does not met certain QoS metrics [7,8]. The difference is that our algorithm does not make the capacity planning decision by itself according to static thresholds, but it triggers a signal to a network manager to revise the logs and make the most convenient decision, based on the fact that a stationary load change has happened.

More related methods to our work are those presented in [9,10]. In [9] the authors make use of wavelets on attempts to detect changes in network measurements for the purpose of anomaly detection. The difference with our work is that we do not desire to detect anomalies (so we remove potentially anomaly data from our datasets, see Section 2) but to detect stationary changes in the network load, i.e. that the patterns of usage, the number of users, etc. have changed. On the other hand, [10] presents an adaptive sampling algorithm to enhance the traffic load measurements. This algorithm improves the results of load change detectors when applied to the measurement step, but does not introduce any novelty in the change detection mechanisms state of the art.

A brief description of our algorithm follows. First, clustering techniques are applied in order to find groups where the intra-group mean value is the same but the mean values between groups are different. To test whether the means are different or not, we apply the Behrens-Fisher methodology (we make no assumption about the covariance matrices), after testing that the data is indeed multivariate normal. The rest of the paper is structured as follows: Section 2 describes the dataset and Section 3 presents the methodology and addresses

the main characteristics of the applied techniques. In Section 4 our online algorithm is described. Section 5 presents the results and Section 6 concludes the paper. Finally, future work is outlined in Section 7.

2 Data Set

We use MRTG [11] measurements from a set of links of the Spanish National Research and Education Network (NREN) RedIRIS[1]. We have collected MRTG logs for the traffic traversing the incoming and outgoing interfaces of several Points of Presence (POP) of the RedIRIS network. With a time granularity of five minutes, we have obtained 288 values for each day. In order to make this sample more manageable, we have averaged such values in 16 disjoint intervals of 90 minutes. The reasons to choose 90 minutes as the averaging period are manifold: first, there is a slim chance of missing data in the five minutes timescale, which is filtered out by averaging in 90 minute periods. Second, the time of the measurements may not be the same in the different POPs due to clock synchronization issues. A timescale of 90 minutes is coarse enough to circumvent this problem (this reason is also pointed out by [5]). Third, the assumption of normality for Internet traffic holds when there is enough temporal aggregation of the measurements [12,13]. Fourth, we require the day duration to be a exact multiple of the averaging period, in order to divide the days in the same intervals and track the daily pattern of network traffic. This daily pattern reflects intervals of high load in working hours and intervals of low load during night periods. Last, but not the least, there is a trade-off between a large averaging period, as required by the aforementioned reasons, and the precision obtained with a smaller one. We believe 90 minutes is a good compromise, which has also been adopted in other studies [5].

As we do not pursue to detect measurement anomalies, we remove potential abnormal data when preprocessing our dataset. Days where at least one of the 90 minutes intervals have no measurements are removed in order to avoid missing values in the dataset. Holidays and exam periods are also removed, since the measurements come from an educational network.

Note that this preprocessing can be performed on-line because these days are known in advance. Thus, the analyzer can be programmed with the days to be withdrawn from the traffic sample. Our measurements last from the 2^{nd} of February 2007 to the 31^{st} of May 2008. After the preprocessing step, the dataset contains more than 200 samples, each corresponding to a day worth of data that we model with a p-variate normal distribution, where $p = 16$ (16 periods of 90 minutes).

To facilitate the understanding of the relation of the number of the variable with the time period of the day to which it refers, these associations are presented in Table 1.

[1] `http://www.rediris.es/`

Table 1. Equivalence in time of the variables

Number of the variable	Time interval	Number of the variable	Time interval
1	00.00-01:30	9	12:00-13:30
2	01:30-03:00	10	13:30-15:00
3	03.00-04:30	11	15:00-16:30
4	04:30-06:00	12	16:30-18:00
5	06.00-07:30	13	18:00-19:30
6	07:30-09:00	14	19:30-21:00
7	09.00-10:30	15	21:00-22:30
8	10:30-12:00	16	22:30-00:00

3 Methodology

In this section we first present the clustering techniques that have been adopted and then provide a brief introduction to the Behrens-Fisher problem. The selected clustering algorithm was k-means[14], which is a two-step iterative algorithm that finds the clusters by minimizing the sum of the squared distances to a representative, which is called *centroid*. The input to the algorithm is the number of clusters k existing in the dataset (since we always look for two clusters, then $k = 2$). The choice of k-means for our online algorithm is due to the ease of adding a new instance to an existing model. To do this, it is only necessary to compute the distance from the new instance to the existing centroids, and then recompute the centroid for the cluster the new instance is assigned to. Finally, if the centroids have changed, k-means is applied again from a quasi-optimal solution, so the algorithm finds the new centroids faster than the first time. On the other hand, in order to obtain clusters that are adjacent in time (i.e. all samples sequential in time and not out of order) the UNIX initial time of the last sample of each day is included as an additional dimension.

To have statistical foundations that the obtained clusters in the former step are in fact different, we have applied the Generalized Behrens-Fisher Problem (GBFP). The GBFP is the statistical problem of testing whether the means of two normally distributed populations (X_1, X_2) are the same (null hypothesis H_0), for the case of unknown covariance matrices. The assumptions are that $X_i \sim \mathcal{N}_p(\mu_i, \mathbf{\Sigma}_i), i = 1, 2$; i.e. the samples of population i come from a p-variate normal distribution with mean μ_i and covariance matrix $\mathbf{\Sigma}_i$. To solve this problem the Hotelling's Generalized T^2-statistic is used, which is distributed as a central F-distribution under the null hypothesis of equality of means. When the sizes of the populations are not equal, a transformation is needed before computing the T^2-statistic (see Section 5.6 of [15]).

The GBFP assumes that the data comes from normal distributions. In order to trust in the results of the GBFP test, we have to make sure that our data is normal. To this end, we have performed several statistical tests to see whether

the assumption of normality holds for each of the clusters. When testing for multivariate normality, it is necessary to perform tests for univariate normality of each of the dimensions, for bivariate normality in all the possible combinations of two dimensions and for p-variate normality (see for instance [16]). For univariate tests we have used Kolmogorov-Smirnov test, Lilliefors test and the Jarque-Bera test. For the multivariate tests we use the multivariate standard distance and χ^2 plots. Although it is necessary to test the normality assumption before each application of the GBFP test, these tests are lightweight and can be performed on-line very fast. If the normality condition does not hold, the distribution of the T^2-statistic under the null hypothesis may differ from the central F-distribution, and thus the probability of rejecting the null hypothesis when it is actually true would be different (Type I error).

4 Online Algorithm

The flux diagram of our algorithm is depicted in Fig. 1. First, daily traffic is collected (16 samples averaging each one 90 minutes of MRTG data) and the timestamp of the day is added (as the dimension 17), giving raise to a time-series of 17-dimensional vectors, where the first 16 dimensions are assumed to come from a 16-variate normal distribution. Then, clustering is applied to the time-series. If the number of samples per cluster is not enough to apply the

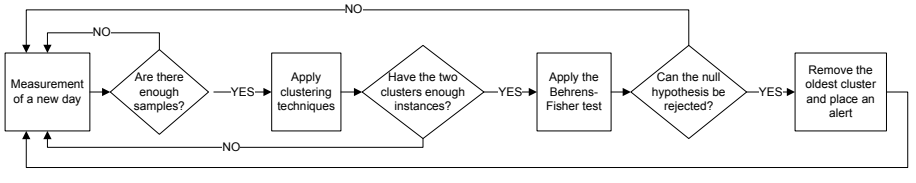

Fig. 1. Flux diagram of the online algorithm

GBPF (less than 17 samples per cluster [15]) we wait for a new day worth of measurements. When both clusters have enough instances, we test for normality and apply the GBFP to the resulting clusters if the normality assumption cannot be rejected. If the GBFP test determines that the null hypothesis of equality of means cannot be rejected, we mark the link as stable and wait for a new day worth of measurements, repeating the process. When the GBFP test shows statistical evidence of a difference in the means at a given significance level α, an alert is sent to the network manager. After the manager is alerted about the possible change in the means, we remove the oldest cluster from the dataset being analyzed and start the algorithm with the newest cluster as input. The results of applying our algorithm to real network measurements are presented in the following section.

5 Results

In this section we present the results of applying our methodology to the measurements of seven links in the RedIRIS network. Table 2 summarizes the number of tests performed and alerts generated. The second and fourth columns show the number of times the Behrens-Fisher testing methodology is applied. This is the number of times that the clustering algorithm was able to form two clusters with enough size to apply the test and the normality assumption held for both sets. It is worth mentioning that the null hypothesis of normality could not be rejected at the significance level $\alpha = 0.05$ for none of the obtained clusters. This supports our initial assumptions about the chosen averaging period (note also that the averaging process reinforces the supposition thanks to the Central Limit Theorem [17]). The third and fifth columns show the number of times an alert signal is sent, i.e. the null hypothesis of equality of means is not verified (again with $\alpha = 0.05$).

Table 2. Results for the online algorithm

University link	Incoming direction		Outgoing direction	
	Number of tests	Number of alerts	Number of tests	Number of alerts
U1	18	9	13	9
U2	13	9	14	7
U3	17	8	12	8
U4	15	8	20	7
U5	13	8	17	9
U6	15	7	11	8
U7	28	8	20	7

As can be seen in Table 2, the main advantage of our online algorithm to network load detection is the reduction in human interventions. This leads to a decrease in the OPEX costs making the network operator save money. The reduction of the human interventions is achieved because our algorithm produces an alert only in case a stationary statistically evident change in the load happens. The rest of the time the link is considered normal, and no intervention from the network manager is required.

Considering the time span of the measurements, our algorithm placed less than 10 alerts (potential network load changes) requiring human supervision in a period of more than 450 days (including holidays). That means a potential stable period between load changes of more than 45 days in average. To illustrate these results, Fig. 2 shows a time-series representation of the obtained groups with statistical evidence of different means. The data showed in that figure refers to the incoming direction of university link U1 for the time interval 12:00-13:30 (Variable 9).

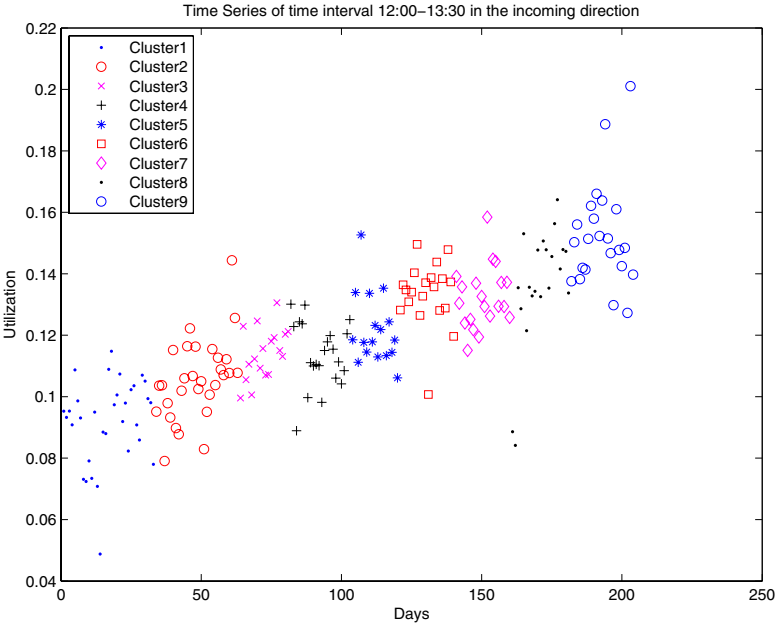

Fig. 2. Time-series plot for time interval 12:00-13:30 showing the different clusters found in the incoming direction of university U1

6 Conclusions

We have presented a new online algorithm for automated detection of network load changes, which has been applied to the Spanish NREN case. The algorithm makes use of well-known statistical techniques to reduce human intervention in network operation. This reduction is achieved by alerting the network manager only when there is statistical evidence of a change in the load, avoiding visual daily inspection of the load graphics for every link in the network. Finally, the capacity planning decision is deferred to the manager supervising the network.

7 Future Work

Several interesting issues remain open for further study. On the one hand, our approach uses a volume model of the network load. It would be interesting to take into account external variables as user demand or access capacity to develop more complex models that increase the accuracy of the detection trigger. On the other hand, it is important to know what limitations introduce the normality assumption and how to cope with them. In this light it would be interesting to know whether the traffic could be modeled with another kind of distribution, maybe using different larger timescales, and in those cases what would be the distribution of the statistic to test for differences in the means.

Acknowledgments. The authors would like to thank the anonymous reviewers for their valuable comments and to acknowledge the support of the Spanish Ministerio de Ciencia e Innovación (MICINN) to this work, under project DIOR (S-0505/TIC/000251).

References

1. Roberts, L.G.: Beyond moore's law: Internet growth trends. Computer (2000)
2. Paxson, V.: Growth trends in wide-area tcp connections. IEEE Network 8(4), 8–17 (1994)
3. Odlyzko, A.M.: Internet traffic growth: sources and implications. In: Proceedings of SPIE, vol. 5247, pp. 1–15 (2003)
4. Pióro, M., Medhi, D.: Routing, Flow, and Capacity Design in Communication and Computer Networks. Morgan Kaufmann Publishers Inc., San Francisco (2004)
5. Papagiannaki, K., Taft, N., Zhang, Z., Diot, C.: Long-term forecasting of Internet backbone traffic. IEEE Transactions on Neural Networks 16(5), 1110–1124 (2005)
6. D'Halluin, Y., Forsyth, P.A., Vetzal, K.R.: Managing capacity for telecommunications networks under uncertainty. IEEE/ACM Transactions on Networking 10(4), 579–588 (2002)
7. Fraleigh, C., Tobagi, F., Diot, C.: Provisioning IP backbone networks to support latency sensitive traffic. In: Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM 2003, vol. 1 (2003)
8. van den Berg, H., Mandjes, M., van de Meent, R., Pras, A., Roijers, F., Venemans, P.: QoS-aware bandwidth provisioning for IP network links. Computer Networks 50(5), 631–647 (2006)
9. Kyriakopoulos, K.G., Parish, D.J.: Automated detection of changes in computer network measurements using wavelets. In: Proceedings of 16th International Conference on Computer Communications and Networks (ICCCN), pp. 1223–1227 (2007)
10. Choi, B., Park, J., Zhang, Z.: Adaptive random sampling for load change detection. In: Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems, pp. 272–273. ACM, New York (2002)
11. Oetiker, T., Rand, D.: MRTG-The Multi Router Traffic Grapher. In: Proceedings of the 12th USENIX conference on System administration, pp. 141–148 (1998)
12. Kilpi, J., Norros, I.: Testing the Gaussian approximation of aggregate traffic. In: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, pp. 49–61 (2002)
13. van de Meent, R., Mandjes, M.R.H., Pras, A.: Gaussian traffic everywhere? In: Proceedings of IEEE International Conference on Communications (ICC), Istanbul, Turkey, vol. 2, pp. 573–578 (2006)
14. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. Wiley, New York (2001)
15. Anderson, T.W., Wilbur, T.: An introduction to multivariate statistical analysis. Wiley, New York (1958)
16. Johnson, R.A., Wichern, D.W.: Applied multivariate statistical analysis. Prentice-Hall International Editions (1992)
17. Durrett, R.: Probability: Theory and Examples. Duxbury Press, Boston (2004)