Performance evaluation of an Online Load Change Detection Algorithm

Felipe Mata and Javier Aracil
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Madrid, Spain
Email: {felipe.mata, javier.aracil}@uam.es

***Abstract*—**In this paper we evaluate an online load change detection algorithm, aimed to identify changes in traffic loads when monitoring Internet links. This online change detector was first introduced in [1] and produces an alert when a sustained and statistically significant change has been detected. Then, the network manager verifies the change and takes action if the change is truly relevant. We show that the behavior of the algorithm with synthetically generated time series is appropriate, and the obtained results are as expected.

***Keywords*—**Change point detection, Algorithm validation, Network monitoring, Knowledge Discovery.

I. Introduction

One important kind of analysis of time series is focused on detecting relevant events that may require immediate attention, which is referred as knowledge discovery. One sort of this knowledge discovery is the detection of change points, which has recently received lots of contributions, both in the form of new algorithms to detect these change points and also in the form of applications of these algorithms to different fields, like econometrics and engineering. In order to identify the strengths and drawbacks of a change detection algorithm, it is crucial to validate its results with appropriately labeled data, i.e. data where the relevant change points are properly identified beforehand and marked in the time series. In this light, the ideal case is a framework where the validation data are obtained from the application domain, and they are afterwards human processed to obtain their relevant events. However, this approach does not cover many possible change cases that were not present in the input data. Instead, a synthetic validation approach allows for more in-depth analysis.

We have developed an online change detection algorithm aimed to identify relevant events in traffic load of the Internet links. This algorithm applies clustering techniques to locate potential change free regions, and statistically sound methodologies to discover whether there is a remarkable change point between them. We described deeply our algorithm and applied it to real network traces showing good performance in a previous work ([1]), but we found it necessary to verify its performance against synthetic data before deployment. Therefore, we have generated different synthetic datasets with the objective of verifying the different properties of our algorithm. It turns out that the overall response of the algorithm to these synthetically generated data is very promising.

The remaining of the paper is structured as follows. Section II is devoted to describe related works in change point detection. Section III reviews our online load change detection algorithm, and its expected data input (i.e. the format and the hypothesis that the data are assumed to fulfill). Next, Section IV describes the synthetically generated data, showing the results of applying them to the load change detector as a function of the significance level α of the statistical tests. In Section V, we analyze the Hotelling's T^2 statistic, which is the statistic used in our hypothesis testing methodology to determine the statistical significance of a change point. Following, we present the results of the algorithm for a fixed significance level α (Section VI). Finally, Section VII concludes the paper.

II. Related Work

Change points are defined as the time positions in the original time series where the local trend is disrupted. Mostly, the problem of detecting change points has been tackled by segmenting the original time series data into portions where the parameters of the chosen model remain unchanged. The most naïve models used in segmentation of time series are linear models. With these segmentation models, the time series are divided into piecewise linear segments, and the change points are located in the time instants where the slope of the linear segment approximations changes. However, this kind of approach usually lacks in either good performance or scalability (i.e. it needs all the data in order to find the segments). In [2] a survey of the different approaches for piecewise linear segmenting is presented, analyzing the aforementioned drawbacks. In addition, the authors present a new algorithm that obtains good performance yet being online (i.e., not needing all the time series to obtain results). To circumvent this weakness, Guralnik et al. present in [3] an algorithm that not only reports changes when the parameters of the model are no longer the same, but also when there is another model more suitable to fit the data (selected from the set of all algebraic polynomials). In addition, more complex models have been also applied to change point detection. For instance, Sharifzadeh et al. [4] use wavelet footprints to detect change points with the same underlying idea of using a polynomial basis, although this approach has the advantage

of scaling well to large datasets because of the compression property of wavelets.

However, these fitted polynomial algorithms (and also other model/parameter change detectors such as [5]) do not use any knowledge of the process that generate the time series. This means that the performance of change detectors can be enhanced for specific applications by properly applying domain knowledge. Therefore, we apply this domain knowledge modeling the samples with a p-variate normal distribution and focusing on changes in the mean, which are the most significant changes for capacity planning tasks of Internet links. Another main difference between our solution and other existing in the literature is that the Behrens-Fisher procedure, which is applied in our algorithm to verify the change points, is equivalent to inspect for change points in p time series at one time (one for each variable), thus enhancing the change point detection.

III. Online Algorithm Description

Our online load change detection algorithm aims to identify sustained and statistically significant change points in network measurements, reporting them to the network manager. The network measurements of interest for the algorithm are load measurements, that can be easily obtained from the Multi Router Traffic Grapher (MRTG). MRTG [6] is a very common tool in network management and monitoring, which reports the average incoming and outgoing transfer rates of each network interface of a network device. Each record is stored in a log file, where the granularity (i.e. time difference between measurements) is often configured to be equal to 5 minutes. This configuration results in 288 values per direction per day. To make this sample more manageable, we average such values in 16 disjoint intervals of 90 minutes, starting at midnight, and in addition we remove holidays to circumvent potential abnormal data. The reasons to choose 90 minutes as the averaging period are reported in [1], being the main one that the Internet traffic can be assumed Gaussian when there is enough time aggregation of the measurements ([7], [8]). Consequently, the load model for a day is a 16-variate normal distribution.

Our methodology then applies k-means (with $k = 2$) and the Multivariate Behrens-Fisher (BF) procedure in an online fashion, as follows. Every time a one-day measurement is completed, it is added to the sample set $\mathcal{S}$. If the cardinal of our sample set is large enough we apply k-means in order to obtain two suitable clusters, i.e. each one with at least 17 samples (we note that we are looking for sustained changes, defining them as change free regions larger than 16 days, so we need $\sharp\mathcal{S} \geq 34$). When we find two suitable clusters, we apply the BF procedure after testing for normality. The BF procedure addresses the statistical problem of testing whether the means of two normally distributed populations $(X^{(1)}, X^{(2)})$[1] are the same (null hypothesis H_0), for the case of unknown covariance matrices. The assumptions are that $X^{(i)} \sim \mathcal{N}_{16}(\vec{\mu}^{(i)}, \mathbf{\Sigma}^{(i)}), i = 1, 2$; i.e. the samples of population i come from a 16-variate normal distribution with mean $\vec{\mu}^{(i)}$ 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 normality assumption does not hold (i.e. the normality tests reject the null hypothesis) the algorithm still goes on and applies the BF test to the populations. However, the network manager is warned about this fact in order to not blindly trust the results of the algorithm. Finally, if the BF test rejects the null hypothesis, an alert is placed to the network manager that indicates a potential change point, and the oldest cluster is removed from the sample set. The flux diagram of Fig. 1 summarizes the algorithm. Interested readers are referred to [1] for a more detailed description of the change load detection algorithm.

[1] we use superscripts between parenthesis to designate populations and subscripts for vector components of the random vectors

IV. Validation of the algorithm

In order to assess the performance of the load change detection algorithm, we have tested it with synthetic data. These synthetic data allow us to verify whether the algorithm is detecting the changes properly. We can do so because we know beforehand where the changes are located. The synthetic datasets generated to test the algorithm can be classified into two different groups, depending on whether they have changes or not. In what follows we describe the datasets generated and show the results of the algorithm performance evaluation. The datasets are N 16-dimensional normal distributed vectors[2], with N = 9000, which is large enough to assess the validity of the obtained results (note that a sample of N = 9000 is equivalent to analyzing approximately 25 years of data in our algorithm).

A. Datasets with no changes

We have generated four datasets with no changes, i.e. having all the samples within the dataset the same mean vector. Even in this case, there is always the chance of detecting a change anyway, thus having False Positives (FP) alarms. These FP can be controlled with the significance level α, which is the probability of rejecting the null hypothesis (that is, detecting a change) even though there is no change in the data (Type I Error). The purpose of these datasets is to evaluate the FP rate under no changes, which asymptotically must approach the probability of Type I Error.

$$\begin{aligned} \mathrm{P}(\text{Type I Error}) &= \mathrm{P}(\text{reject } H_0 | H_0 \text{ is true}) = \alpha \\ &= \lim_{M \to \infty} \frac{\#\text{ of rejections}}{M}, \end{aligned} \tag{1}$$

where M is the total number of tests performed with datasets that fulfill H_0.

[2] all the vector components are independent of each other

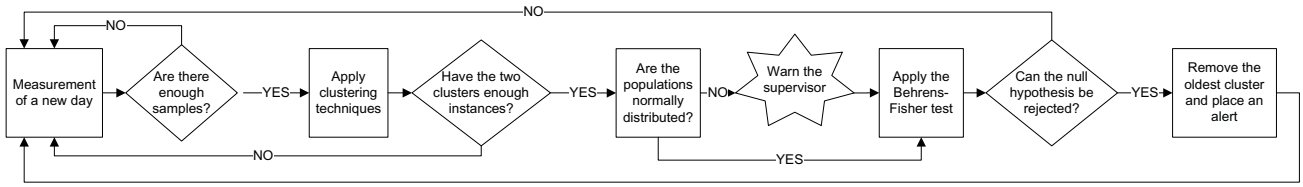

Fig. 1. Flux diagram of the online algorithm.

TABLE I
DATASETS GENERATED WITH NO CHANGES.

Dataset	Description
All Equal (AE)	All vector components have the same mean and variance
Means (M)	Each vector component has a different mean, but their variances are the same
Variances (V)	Each vector component has the same mean, but different variance
Means Variances (MV)	Each vector component has different mean and variance

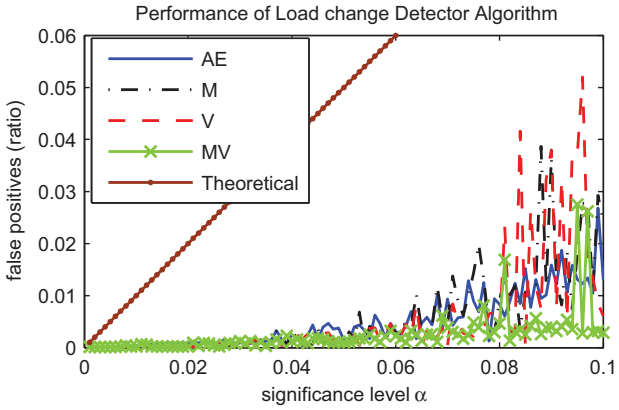

Fig. 2. False positives ratio in datasets with no changes.

1) Description of the datasets: The four datasets generated without changes in their means are obtained through four different affine transformations on four different random samples of N realizations distributed according to a standard 16-variate normal distribution. The applied transformations have been chosen in order to obtain four datasets with the characteristics that are summarized in Table I.

2) Results: We have measured the false positives ratio (FPR) given by (1) for different significance levels α. The results are presented in Fig. 2, which shows the FPR of each dataset versus the significance level used in the tests, also with the theoretical FPR (that equals α). The FPR remains almost negligible for significance levels smaller than $\alpha = 0.06$. Thus, we have a large interval of possible significance levels with good performance. Significance levels above 0.06 experiment an increment in the FPR, but also in this region the FPR of the algorithm when applied to these datasets is smaller than the theoretical one. The differences in the performance of the algorithm for the four different datasets are not relevant, because these differences are mainly due to random number generation issues (we have confirmed this by applying different transformations to the same random generated sample).

B. Datasets with staggered increments

As the aim of the algorithm is to detect changes in the load, after confirming that there is a low ratio of false positives, a validation with controlled changes follows. Thus, we have generated two different datasets with staggered increments of duration one and three months, i.e. the distribution of the samples remain the same for one (three) month(s), and after that, the mean is increased. We note that this kind of growth is the most significant for the capacity planning task, because linear increments are easily tracked by classical time series analysis, so a forecast of upgrading times when the changes are linear is straightforward. This is accomplished by fitting a time series model to the data (for instance an AutoRegressive Integrated Moving Average model [9]) and then predicting when the time series will be above a given threshold ([10]) where the Quality of Service (QoS) of the link might be compromised. Therefore, detecting staggered increments in a timely fashion is crucial for network operators, because the reduction in QoS delivered to its customers adversely affects the operator's reputation.

1) Description of the datasets: The growth rate for the monthly staggers is chosen such that effective annual growth is around 90%, which is in accordance with popular reports about the Internet traffic growth ([11]). Thus, the monthly growth is approximately 6%. The quarterly growth has also been set to approximately 6%, on attempts to make the obtained results comparable, i.e. we have longer periods without changes in the quarterly growth dataset, but the size of the staggers (which are the relevant facts to detect changes) are the same in both time series. Finally, the theoretical number of changes that should be detected with the algorithm in the Monthly Increments (MI) dataset is 300 and in the Quarterly Increments (QI) dataset is 100.

2) Results: In Fig. 3 we show the number of detected changes on the MI data as a function of the significance level of the performed tests. This figure shows very promising results. The number of detected changes is in the range 295-300, while the correct value is 300. In addition, the number of false negatives is small for all the significances tested.

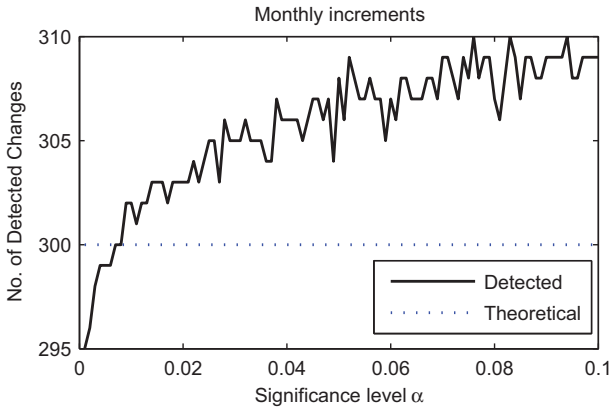

Fig. 3. Detected changes in Monthly Increments dataset.

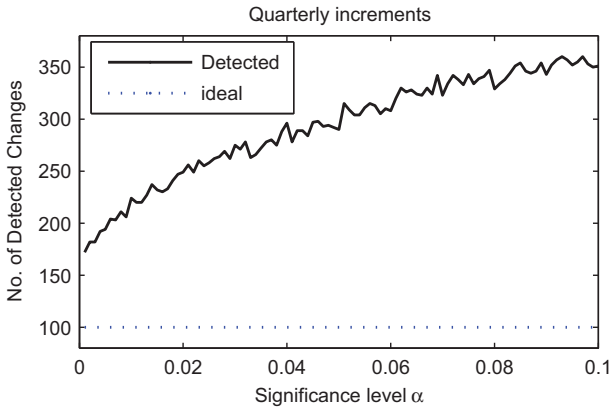

Fig. 4. Detected changes in Quarterly Increments dataset.

Figure 4 presents the same information but for the QI data. Here the performance has been reduced. There is no significance level at which we detect exactly the same number of changes that are theoretically in the dataset. In addition, the false positives have enlarged, being now greater than 50. Above significance values greater than 0.06 we detect more than 300 changes, meaning that every theoretical change we alert for 3 detected changes. We will shed light on the causes of this misidentification in Section VI by inspecting the results at a fixed significance level.

V. Analysis of the Hotelling's T^2 Statistic

We now analyze the Hotelling's T^2 statistic, in order to apply our conclusions in the following section. The Hotelling's T^2 statistic for the multivariate Behrens-Fisher problem is as follows:

$$T^2 = N \frac{Y S_y^{-1} Y^t}{N-1} \frac{N-p}{p}, \tag{2}$$

where N is the number of samples that were used to compute Y, p is its dimension and S_y is the estimated covariance matrix. Y is a p-dimensional vector $Y = (Y_1, Y_2, \ldots, Y_p)$ of the means of the differences between both populations (however, if the populations have not the same size, a transformation is needed before computing the means, see [12]). This statistic follows a Snedecor's F distribution with p and $N-p$ degrees of freedom under H_0.

The term $Y S_y^{-1} Y^t$ is a quadratic form of the p vector components of the random vector Y. As we are using synthetic data, we can approximate with the covariance matrix used to generate the samples in what follows. This matrix has been chosen to be diagonal (remember that the vector components are independent). This implies that the quadratic form is the weighted sum of the square of all the vector components (being the weights given by the elements of the diagonal covariance matrix). In the simplest case, all the vector components have the same variance, so the covariance matrix is a multiple of the identity matrix. Assuming equal all the vector components of Y, this gives us

$$\begin{aligned} T^2 &= N \frac{Y S_y^{-1} Y^t}{N-1} \frac{N-p}{p} \approx N \frac{Y \frac{1}{\sigma^2} \mathbb{I}_p Y^t}{N-1} \frac{N-p}{p} = \\ &= \frac{N}{N-1} \frac{N-p}{p} \sum_{i=1}^{p} \frac{y_i^2}{\sigma^2} \approx \frac{N}{N-1} \frac{N-p}{p} \frac{p y^2}{\sigma^2} = \\ &= N \frac{N-p}{N-1} \frac{y^2}{\sigma^2}. \end{aligned} \tag{3}$$

If we fix the significance value α, we are comparing the value obtained from (2) against a value that is a function of N (given that the dimension of the random vector p is also fixed). This function is the 1-α percentile of the F distribution with p and $N-p$ degrees of freedom ($F_{p,N-p}^{1-\alpha}$). We reject H_0 if the T^2 statistic value is greater than the value of the function evaluated in that N, which is equivalent to

$$\frac{y^2}{\sigma^2} > \frac{F_{p,N-p}^{1-\alpha}}{N} \frac{N-1}{N-p}. \tag{4}$$

VI. Analysis at fixed significance level

In this section, we further inspect the synthetic data presented in Section IV, but with a fixed value for the significance level. The value selected for the significance level is $\alpha = 0.05$, as it is the most commonly used value. By making the significance level fixed we can present graph plots of the clusters found and inspect the reported change points. On those graphs, we plot the values of the projection in one vector component, using different color-marker schemes to differentiate the change free regions according to the results of the algorithm. In addition, we mark with a straight line the mean of all the values within a change free region, making it easier to judge the validity of the reported change points. As the amount of points generated for each vector component is humongous, we will focus on certain regions of the plots that we have found to be relevant for the validation.

A. Datasets with no changes

This subsection is devoted to inspect the datasets generated with no changes. In what follows, we focus on the AE dataset, as we have found it to be representative of all the datasets generated with no changes.

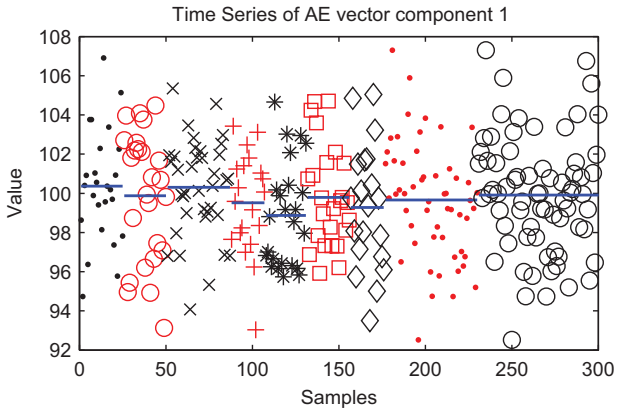

Fig. 5. Time Series representation of the change free regions for the first 300 samples of the 1^{st} vector component of the AE dataset.

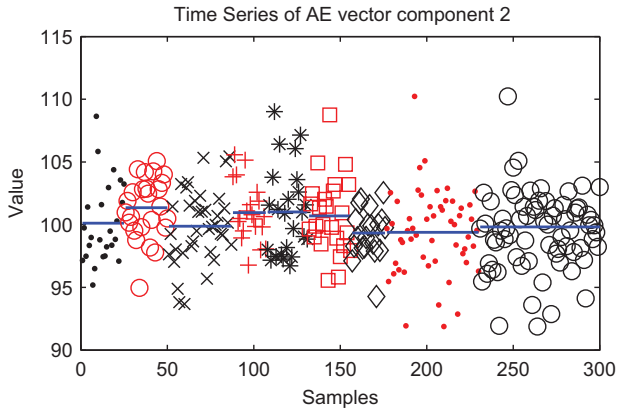

Fig. 6. Time Series representation of the change free regions for the first 300 samples of the 2^{nd} vector component of the AE dataset.

In Fig. 5, we show the change free regions found by the algorithm in the first 300 samples of the AE dataset. Although the samples are concentrated around the true mean (100), the algorithm detected some change points. This happens because we are applying a statistical test, whose confidence level can be interpreted as the rate of false positives in the limit. Therefore, although a perfect algorithm would have detected no changes in this dataset, it is a normal situation when applying statistical tests to have some false positives due to the confidence level.

The change points reported by the algorithm in this dataset can be due to the following reasons:

- The algorithm found one cluster with mean above the theoretical followed by a cluster with mean under the theoretical (or vice versa). This can be easily seen between the first two change free regions in Fig. 5.
- The weighted sum of the differences in all the vector components is above $F^{1-\alpha}_{p,N-p}$ (Section V). To illustrate this fact, we present in Fig. 6 the same zoom area for vector component 2. The differences between the last two change free regions on Fig. 5 and Fig. 6 (the dots (·) around sample 200 and the circles (○) on its right) are very small, but the addition of these differences through all the variables motivates reporting a change point.

B. Datasets with staggered increments

As was described in Section IV-B, these datasets are designed to be invariant both in mean and variance for a fixed period of time after which the value of the mean is increased. Thus, in these regions without changes we are in the same case as in the AE dataset. We therefore inspect each stair of the dataset from the point of view used in Section VI-A.

1) Monthly increments dataset: The clusters in the final samples of this dataset (sample 8000 and above) are easily identified by the algorithm, as the differences between those clusters are big enough due to the increment by percentages in each theoretical change point. Thus, we will zoom in the beginning of the dataset and focus on the first samples (sample 120 and under). This region is depicted in Fig. 7, where

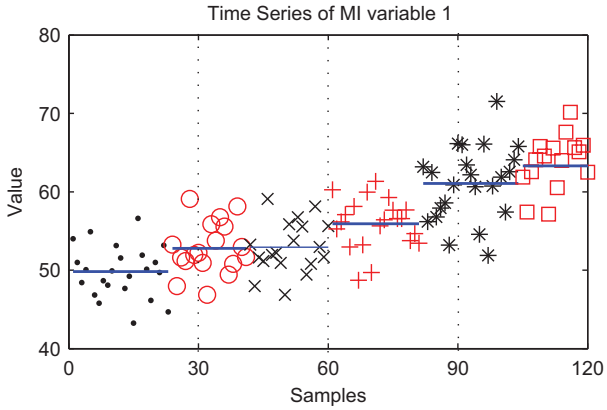

Fig. 7. Zoom to the first 120 samples of the 1^{st} vector component of the MI dataset with delimitation lines for the theoretical change points.

we have placed vertical lines in the time instants where the theoretical change points are located.

As can be seen in the figure, the variance of the sample is big enough to make samples in different theoretical change free regions (therefore with different means) to be indistinguishable in some cases. For instance, take a look in the first change free region (under sample 30). The circle (○) samples in this region are generated with the same mean as the dot (·) ones. However, these circle samples resemble more to those circle samples in the second change free region (between samples 30 and 60) than to the dot ones with the same theoretical mean. This is detected by the algorithm through the clustering technique, which divides the first region before the theoretical change. As the difference between the means is truly significant, the Behrens-Fisher procedure detects it and a change point is reported between these clusters. That is what makes the algorithm to misinterpret the true change point between those regions, which we have confirmed to happen also in other instants of the dataset. This rationale explains all the false positives detected by the algorithm, that under small variance samples or with a more restrictive significance value would not have been detected. However, if we pay

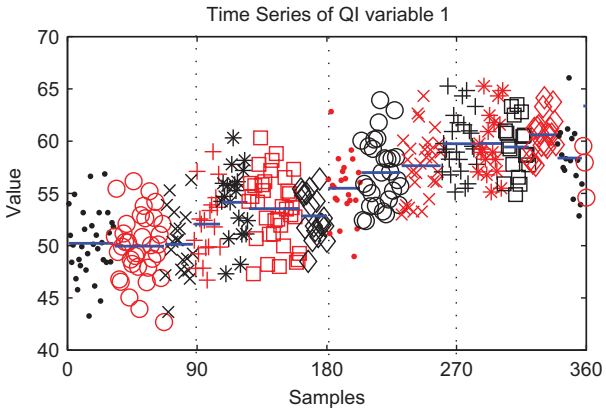

Fig. 8. Zoom to the first 360 samples of the 1^{st} vector component of the QI dataset.

attention to the second change free region, we find that there are not significant differences between the two clusters found by the algorithm when inspecting them visually. Remember form Section VI-A that the detected change point between these two clusters is also due to the differences in the means of the remaining vector components, although apparently in this component there is no change.

2) Quarterly increments: In this section, we deal with the staggered synthetic data whose increments happen every three months (90 samples). For the same reason that in the MI dataset, we will zoom in to the first samples, because there the samples are more concentrated and it is difficult to assess the validity of the algorithm without this zoom. In Fig. 8 we have zoomed in to the first 360 samples, and represented the change free regions found by the algorithm in conjunction with their means and the theoretical change points.

In that figure it can be easily seen that in each theoretical change free region, our algorithm reported several change points. The reason for the detection of these extra change points is the same pointed out in Section VI-A, as the extra change points are detected within a theoretical change free region, where the mean and the variance remain constant (same as AE dataset). On the other hand, there are some theoretical change points not reported by the algorithm (for instance the one in sample 270). The reason for the misidentification of some theoretical change points was described in the previous subsection for the MI dataset. As the samples have a relatively large variance (compared to their mean) in this region, this leads to samples of one theoretical change free region that resemble more to those of adjacent regions than to the samples on its own region. This similarity is detected by the clustering algorithm, and the fact that there is actually a difference between them is finally confirmed by the statistical procedure.

VII. Conclusions

In this paper, we have assessed the performance of an online change load detector aimed to identify changes in the load of the Internet links. The algorithm was presented in [1], and uses a p-variate normal distribution to model the Internet traffic load within a day ($p = 16$). The measurements are input to the algorithm in a per day basis, after a preprocessing step where among other tasks abnormal data is removed (e.g. holidays). The algorithm then places alerts to a network manager when there is statistical evidence of sustained load changes. The supervisor is also warned when the hypothesis of the algorithm are not satisfied by the data. We showed the behavior of the algorithm by running it against real network measurements obtained from the Spanish National Research & Education Network RedIRIS[3] (see [1] for a description of these results).

However, an exhaustive validation against labeled data was missing. Therefore we have generated synthetic data fulfilling algorithm's hypothesis and assessed the validity of the obtained results. When using this data, the false positives ratio is very low (Fig. 2) and the accuracy identifying change points with staggered data is remarkable (Fig. 7). We envisage that these are the more important change points in network monitoring for capacity planning, because linear changes are easily tracked by classical time series analysis, so a forecast of capacity planning update time instants is straightforward.

VIII. Acknowledgments

The authors would like to acknowledge the support of the Spanish Ministerio de Ciencia e Innovación (MICINN) to this work under the FPU fellowship program.

References

[1] F. Mata, J. Aracil, and J. L. García-Dorado, "Automated Detection of Load Changes in Large-Scale Networks," in *International Workshop on Traffic Monitoring and Analysis*, May 2009, pp. 34–41.
[2] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "An online algorithm for segmenting time series," in *Proceedings of IEEE International Conference on Data Mining*, 2001, pp. 289–296.
[3] V. Guralnik and J. Srivastava, "Event detection from time series data," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, 1999, pp. 33–42.
[4] M. Sharifzadeh, F. Azmoodeh, and C. Shahabi, "Change detection in time series data using wavelet footprints," *Lecture Notes in Computer Science*, vol. 3633, p. 127, 2005.
[5] V. Puttagunta and K. Kalpakis, "Adaptive methods for activity monitoring of streaming data," in *Proceedings of the International Conferences on Machine Learning and Applications*, 2002, pp. 197–203.
[6] T. Oetiker and D. Rand, "MRTG: The Multi Router Traffic Grapher," in *USENIX Conference on System Administration*, 1998, pp. 141–148.
[7] J. Kilpi and I. Norros, "Testing the Gaussian approximation of aggregate traffic," in *ACM SIGCOMM Workshop on Internet Measurment*, 2002, pp. 49–61.
[8] R. van de Meent, M. Mandjes, and A. Pras, "Gaussian traffic everywhere?" in *IEEE International Conference on Communications*, vol. 2, June 2006, pp. 573–578.
[9] K. Papagiannaki, N. Taft, Z.-L. Zhang, and C. Diot, "Long-term forecasting of Internet backbone traffic," *IEEE Transactions on Neural Networks*, vol. 16, no. 5, pp. 1110–1124, Sept. 2005.
[10] P. J. Brockwell and R. A. Davis, *Time series: theory and methods*, ser. Springer Series in Statistics. Springer, 1991.
[11] A. M. Odlyzko, "Internet traffic growth: sources and implications," *Proceedings of SPIE*, vol. 5247, pp. 1–15, 2003.
[12] T. W. Anderson, *An introduction to multivariate statistical analysis*. Wiley New York, 1958.

[3] http://www.rediris.es/