Trend Forecasting based on Singular Spectrum Analysis of Traffic Workload in a Large-Scale Wireless LAN ⋆

George Tzagkarakis, Maria Papadopouli and Panagiotis Tsakalides

Department of Computer Science, University of Crete

Institute of Computer Science, Foundation for Research and Technology-Hellas P.O. Box 1385, 711 10 Heraklion, Crete, Greece

Abstract

Network traffic load in an IEEE802.11 infrastructure arises from the superposition of traffic accessed by wireless clients associated with access points (APs). An accurate load characterization can be beneficial in modelling network traffic and addressing a variety of problems including coverage planning, resource reservation and network monitoring for anomaly detection. This study focuses on the statistical analysis of the traffic load measured in a campus-wide IEEE802.11 infrastructure at each AP. Using the Singular Spectrum Analysis approach, we found that the time-series of traffic load at a given AP has a small intrinsic dimension. In particular, these time-series can be accurately modelled using a small number of leading (principal) components. This proved to be critical for understanding the main features of the components forming the network traffic.

The statistical analysis of leading components has demonstrated that even a few first components form the main part of the information. The residual components capture the small irregular variations, which do not fit in the basic part of the network traffic and can be interpreted as a stochastic noise. Based on these properties, we also studied contributions of the various components to the overall structure of the traffic load of an AP and its variation over time. Finally, we designed and evaluated the performance of a traffic predictor for the trend component, obtained by projecting the original time-series on the set of leading components.

Key words: Traffic load modelling, traffic load forecasting, singular spectrum analysis, wireless networks

1. Introduction

Wireless networks are increasingly being deployed to provide Internet access in airports, universities, corporations, hospitals, residential, and other public areas. Furthermore, there is a growth in peer-to-peer, streaming, and VoIP traffic over the wireless infrastructures (1; 2). Wireless Local Area Networks (WLANs) have more vulnerabilities and bandwidth/latency constrains than their wired counterparts. The bandwidth utilization at an AP can impact the performance of the wireless clients in terms of throughput, delay, and energy consumption. For quality of service provision, capacity planning, load balancing, and network monitoring, it is critical to understand the traffic characteristics. For this purpose, the design of accurate models of the network and client activity are critical. In addition, the traffic models can assist in detecting abnormal traffic patterns (e.g., due to malicious attacks, AP or client misconfigurations and failures).

One of the most intriguing aspects of the traffic demand modelling in WLANs is its intrinsic multi-level, spatio-temporal nature, namely, the different spatial scales (e.g., infrastructure-wide, AP-level or client-level) and time granularities, such as packet-level, flow-level and session-level. While there is a rich literature characterizing traffic in wired networks (4; 5; 6), there are only a few studies available examining wireless traffic load (1; 2; 9; 10; 28; 29; 30; 31).

In a recent work (7; 8), two key structures in a WLAN, namely, the session of a client and the traffic flows generated within that session by that client, were modelled in both spatial and temporal dimensions, and their dependencies and interrelations were examined.

Admission control and load-balancing algorithms can benefit from short-term forecasting of traffic load at hotspots. However, such traffic predictions are challenging and suffer from large prediction errors due to the high variability (9; 10). These studies analyzed the traffic at

⋆ This work was supported by the Greek General Secretariat for Research and Technology under Programs ΠΕΝΕΔ-Code 03ED69 Regional of Crete, Crete-Wise KP-18 (ΚΠΣ 00126) and 05NON-EU-238, and the European Commission (MIRG-CT-2005-029186).
Contact author: Maria Papadopouli (mgp@ics.forth.gr, maria@csd.uoc.gr).

hotspot APs and evaluated traffic forecasting algorithms based on recent history, periodicities, and number and type of flows. Specifically, the traffic load at APs was modelled using variants of the Moving Average and Autoregressive Moving Average models, resulting in simple forecasting methods. This paper extends these research efforts. To the best of our knowledge, this is the first study in characterizing statistically wireless traffic load using non-linear time-series analysis techniques.

This research analyzes data collected using the Simple Network Management Protocol (SNMP) from a large-scale wireless infrastructure (11) using a lightweight acquisition methodology. SNMP is the most widely-available monitoring service in wireless platforms. Any AP in the market supports monitoring using SNMP, so it is important to understand how much operators and researchers can learn from SNMP data. Furthermore, this type of data is the most appropriate one to understand daily and long-term trends in the usage of wireless networks. This paper makes use of SNMP data for analyzing traffic characteristics, such as total load and periodicities.

To achieve a deeper understanding of the main features of traffic measurements, we employ a non-linear time-series analysis (12; 3). At the same time, due to the complicated structure of a traffic series, traditional algorithms of non-linear analysis may not estimate reliably the analyzed time-series. However, after filtering out a high-frequency component, which can be considered as a noisy part, we expect to obtain a more accurate estimation of the embedding dimension of the underlying process. Motivated by this observation, in this study, we analyze traffic series by decomposing them in two components, namely, a low-frequency and a high-frequency one, using Singular Spectrum Analysis (SSA).

SSA (14) belongs to the general category of Principal Component Analysis (PCA) methods (13), which apply a linear transformation of the original data space into a feature space, where the data set may be represented by a reduced number of "effective" features while retaining most of the information content of the data. The SSA method is very efficient for the analysis of time-series corresponding to an arbitrary process. In a recent work (15), SSA was used to analyze the dynamics of traffic obtained at an intermediate-scale wired LAN. To the best of our knowledge, this is the first study that applies SSA on the analysis of traffic from a WLAN.

This paper employs SSA to explore the intrinsic dimensionality and structure of the time-series corresponding to the traffic load at a given AP, using data collected from a campus-wide WLAN infrastructure. To investigate the nature of this dimensionality, we introduce the notion of *eigenloads*. Derived from the implementation of SSA on a given traffic load series, an eigenload is a time-series that captures a particular source of temporal variability. Each traffic load series can be expressed as a weighted sum of eigenloads, where the weights are proportional to the extend to which each eigenload is present in the given traffic load series.

We show that traffic eigenloads in a WLAN fall into two natural classes:

- *deterministic eigenloads*, which capture the slow-varying trends in the traffic load series, and
- *noise eigenloads*, which account for traffic fluctuations appearing to have relatively time-invariant properties.

By categorizing eigenloads in this manner, we obtain a significant insight into the intrinsic properties of the traffic load series. In particular, we find that each time-series can be well approximated by only a small number of eigenloads, which constitute its "feature set". Furthermore, these features vary in a predictable way as a function of the amount of traffic carried in the time-series. We show that the largest traffic load series, i.e., the series with the highest mean traffic load, are primarily deterministic. On the other hand, traffic load series of moderate size are generally comprised of noisy features.

Motivated by the observation that the deterministic part of the traffic load series presents a slow variation in time and carries the main part of the information content, we design a predictor that performs trend forecasting at a larger than an hourly time-scale. This forecasting algorithm is based on the modelling of the traffic-series using a linear model of order p, whose coefficients (weights) are estimated using the Normalized Least Mean Squares (NLMS) approach.

This work extends our earlier research (30) with the following significant contributions:

- the design and evaluation of the performance of a traffic predictor based on our proposed models
- the validation of the proposed modeling approach by applying it on the uploading and downloading traffic separately.

The paper is organized as follows: Section 2 describes the wireless infrastructure at the University of North Carolina at Chapel Hill (UNC) and the data acquisition process. In Section 3, we present the basic concept of the SSA approach. We apply this method on our traffic measurements and analyze the leading components, which are responsible for the main part of the network's traffic, and the residual components, which can be represented as irregular variations of the data. Section 4 provides a statistical modelling for a set of traffic load series, then applies SSA to these time-series, and presents the low-dimensionality property. Section 5 presents the classification of the eigenloads in two classes and focuses on the characteristics of the decomposition of traffic load series into their constituent eigenloads. Section 6 describes the design of a traffic predictor for the trend component of the original time-series and evaluates its performance by applying it on the time-series containing the amount of bytes received and sent, as well as, the aggregate amount of traffic, from all clients that were associated with a particular AP. Finally, Section 7 summarizes our main results and discusses future work plans.

2. Background

The IEEE802.11 infrastructure at UNC provides coverage for the 729-acre campus and a number of off-campus administrative offices. The university has 26.000 students, 3.000 faculty members and 9.000 staff members. Undergraduate students (16.000) are required to own laptops, which are generally able to communicate using the campus wireless network. A total of 488 APs were part of the campus network at the start of our study. These APs belong to three different series of the Cisco Aironet platform: the state-of-the-art 1210 Series (269 APs), the widely deployed 350 Series (188 APs) and the older 340 Series (31 APs).

The data was collected using SNMP for polling every AP on campus every five minutes. First, the data collection system was implemented using a nonblocking SNMP library for polling each wireless access point (AP) precisely every five minutes in an independent manner. This eliminates any extra delays due to the slow processing of SNMP polls by some of the slower APs. The system ran in a multiprocessor system and the CPU utilization in each of the three processors we employed never exceeded 70%. Second, our characterization of the workload of the APs is derived only from those clients associated with the AP at polling time.

The data collection took place between 9:09 a.m. September 29th, 2004 and 12:00 a.m. November 30th, 2004. The total number of polling operations during the 63 days was 8.247.479. The data collection system ran flawlessly for the entire period, but APs were sometimes unresponsive. This is generally due to maintenance down-times, reboots, or overloads. If an AP did not respond to a poll, the data collection system tried again 5 *sec* later (and if necessary, again after 10 *sec* and 15 *sec*). It is therefore unlikely that datagram losses created holes in our dataset.

Based on the SNMP trace for each AP, we produced a time series of its traffic load at hourly intervals. This traffic is the total amount of bytes received and sent from all clients that were associated with the AP at that time interval. In the rest of the paper, depending on the mathematical expression, we will use two notations for these time series. Specifically, the traffic of the AP i during the h-th hour of day d ($h \in \{1, \dots, 24\}$, $d \in \{1, \dots, 63\}$), that corresponds to time t, is $T_i(h, d) = X_i(t)$.

3. Singular Spectrum Analysis of a time-series

Singular Spectrum analysis (SSA) is a method suitable for extracting information from short and noisy time series. It unravels the information embedded in the delay-coordinate phase space by decomposing the sequence into elementary patterns of behavior in time and spectral domains, that help separating the time series into statistically independent components, which can be classified as trends and deterministic oscillations (or noise).

SSA looks for structures in a time series by doing an eigendecomposition of the so-called lagged covariance matrix. This approach is useful in non-linear system analysis, because as opposed to other time-series analysis techniques, we do not have to choose the structure functions *a priori*, but instead, the data lets themselves to choose the temporal structures.

Time-series corresponding to wireless traffic load are often short and contain typically peaks on top of a more regular background. Besides, these series often have both regular (periodic) and irregular (noisy) aspects, which may be present in different spatial and temporal scales. Thus, the need for combining a deterministic with a stochastic modelling approach is necessary, motivating the use of the SSA approach. The following paragraphs describe the modelling process step by step and apply it on randomly selected time-series corresponding to several hotspot APs of our dataset.

3.1. *Introduction to SSA*

The SSA is applied to the analysis of time-series corresponding to an arbitrary signal $x(t)$, with $t > 0$. The standard SSA consists of four main steps:

(i) Transformation of the one-dimensional time-series into a trajectory (Hankel) matrix
(ii) Singular Value Decomposition (SVD) of the Hankel matrix
(iii) PCA and selection of the dominant features by grouping the SVD components
(iv) Reconstruction of the original time-series using the selected features (inverse Hankelization by diagonal averaging)

Let $X = \{x_j\}_{j=1}^{N}$ denote the samples of the time-series and L ($1 < L < N$) be an integer, indicating the (caterpillar) window length. The transformation step forms $K = N - L + 1$ lagged vectors $X_k = \{x_k, \dots, x_{k+L-1}\}^T$, $1 \leq k \leq K$. The trajectory Hankel matrix of the time-series X is of dimension $L \times K$ and has the following form:

$$\mathbf{H} = [X_1 \; X_2 \; \cdots \; X_K] \; . \tag{1}$$

The trajectory space is defined as the linear space spanned by the columns of $\mathbf{H}$.

After the above Hankelization process, the SSA method performs an SVD of the matrix $\mathbf{C} = \mathbf{H}\mathbf{H}^T$. Let $\lambda_1 \geq \ldots \geq \lambda_L$ be the eigenvalues of $\mathbf{C}$, which give the energy attributable to the respective principal component, and $r = \max\{i : \lambda_i > 0\}$. Let $U_1, \dots, U_r$ denote the corresponding eigenvectors (principal components) and $V_j = \mathbf{H}^T U_j / \sqrt{\lambda_j}$, $j = 1, \dots, r$, the set of factor vectors, which capture the temporal variation common to all lagged vectors along the j-th principal axis. We refer to the set $\{V_j\}_{j=1}^{r}$ as the set of *eigenloads* of X.

Since the principal axes are in order of contribution to the overall energy, V_1 captures the strongest temporal trend common to all lagged vectors, V_2 captures the next strongest trend and so on. If we denote $\mathbf{H}_j = \sqrt{\lambda_j} U_j V_j^T$, the trajectory matrix $\mathbf{H}$ can be written as

$$\mathbf{H} = \mathbf{H}_1 + \cdots + \mathbf{H}_r \; . \tag{2}$$

By applying the inverse Hankelization process on each matrix $\mathbf{H}_j$, we obtain an approximation X^j of the original series X.

Once the expansion given by (2) has been completed, the third step of the SSA method consists of partitioning the set of indices $\mathcal{I} = \{1, \ldots, r\}$ into s disjoint subsets, where the value of s depends on the specific application. Let $\mathcal{I}_1 = \{i_1, \ldots, i_m\}$ be the first subset of indices, and $\mathbf{H}^{\mathcal{I}_1} = \mathbf{H}_{i_1} + \cdots + \mathbf{H}_{i_m}$ to be the approximation of the trajectory matrix $\mathbf{H}$ based on the indices of $\mathcal{I}_1$. Similarly, we have an analogous decomposition corresponding to each subset $\mathcal{I}_k$, $k = 2, \ldots, s$. Thus, we obtain the final decomposition of the initial Hankel matrix $\mathbf{H}$:

$$\mathbf{H} = \mathbf{H}^{\mathcal{I}_1} + \cdots + \mathbf{H}^{\mathcal{I}_s} \ . \tag{3}$$

The last step of the SSA is the application of an inverse Hankelization process on the approximation matrix $\mathbf{H}^{\mathcal{I}_k}$, $k = 1, \ldots, s$, to approximate the initial time-series. This process is simply performed by averaging the elements of $\mathbf{H}^{\mathcal{I}_k}$, which are placed on the same anti-diagonal, that is, the elements $h_{ij}^{\mathcal{I}_k}$ with $i + j =$ constant. Let $X^{\mathcal{I}_k}$ denote the time-series reconstructed using the matrix $\mathbf{H}^{\mathcal{I}_k}$. Then, the j-th component of $X^{\mathcal{I}_k}$, $x_j^{\mathcal{I}_k}$, is given by: $x_j^{\mathcal{I}_k} =$ mean{elements of $\mathbf{H}^{\mathcal{I}_k}$ which are placed on the j-th anti-diagonal} [1] . Thus, the result of SSA is an expansion of the original time-series into a sum of s series,

$$X = X^{\mathcal{I}_1} + \cdots + X^{\mathcal{I}_s} \ , \tag{4}$$

where $X^{\mathcal{I}_k}$ is the time-series reconstructed using the matrix $\mathbf{H}^{\mathcal{I}_k}$. For instance, the case $s = 2$ can be interpreted as a problem of separating a signal from a noise component. The performance of the method mainly depends on two parameters, namely, the selection of the window length L and the partitioning of the positive eigenvalues. In the following, we describe procedures for the specification of these parameters, when using SSA for WLAN traffic workload analysis.

3.2. *Estimation of the window length L*

The selection of a suitable window length, L, is crucial for an increased accuracy of the SSA. The value of L is computed, such that the points of different lagged vectors, X_k, X_l $(k \neq l)$, can be considered as linearly independent. In our context, for an arbitrary AP, the window length L is chosen to be equal to the correlation length (time lag), i.e., when the sample auto-correlation function

$$C(L) = \frac{\sum_{j=1}^{N} (x_{j+L} - \bar{x})(x_j - \bar{x})}{\sum_{j=1}^{N} (x_j - \bar{x})^2} \ , \tag{5}$$

crosses for the first time the confidence interval corresponding to the white Gaussian noise. In this case, the lagged vectors of length L can be considered to be independent, which enables each vector to be analyzed separately. In (5), $\bar{x}$ denotes the arithmetic mean of the time-series X.

[1] The first anti-diagonal is simply the element $h_{11}^{\mathcal{I}_k}$.

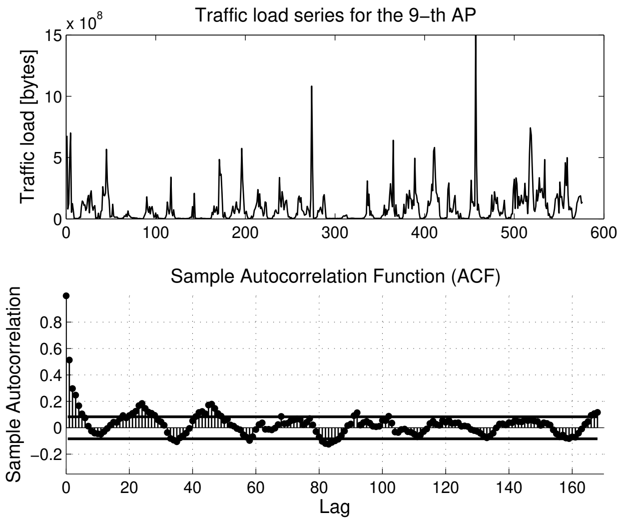


Fig. 1. Traffic load series and sample auto-correlation function for $X_9(t)$.

Fig. 1 presents the aggregate traffic load series and the values of the sample auto-correlation function as a function of the window length L (time lag), together with the confidence interval corresponding to the white Gaussian noise, for an AP of our dataset. As shown, the auto-correlation function first crosses the confidence interval for $L = 7$, that is, the selected window length should be equal to 7.

In the following two subsections, we describe the procedure for partitioning the set of the r eigentriples $\{\lambda_j, U_j, V_j\}_{j=1}^{r}$ into s disjoint subsets. For convenience, we focus on the case that the eigentriples are divided in two classes, namely, the principal and the residual eigentriples (i.e., $s = 2$).

3.3. *Analysis of leading components*

As it was mentioned before, the eigenvalues given by applying an SVD on the trajectory matrix $\mathbf{H}$ can be used to select a set of *feature components* for the reconstruction of the original time-series. In particular, the ratio

$$R_i = \frac{\lambda_i}{\sum_{j=1}^{L} \lambda_j} \tag{6}$$

is used to estimate the energy contribution (in decreasing order) of the i-th principal component in the analyzed time-series, which can be represented as the fraction of the information content related to that (single) component.

Fig. 2 shows the contribution of the eigenvalues corresponding to the aggregate traffic series of the 9-th AP in our dataset, for two different window lengths, namely, the window length $L_1 = 7$ given by the auto-correlation function (5), and $L_2 = 14$. This information permits the estimation of the number of principal components which effectively contribute to the information content of the time-

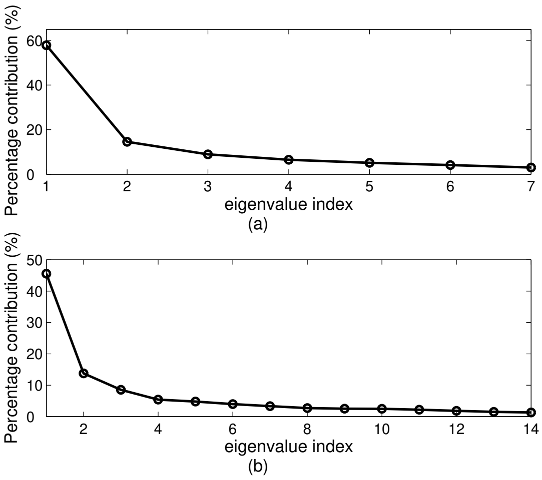


Fig. 2. Percentage contribution of the eigenvalues for $X_9(t)$: (a) $L = 7$, (b) $L = 14$.

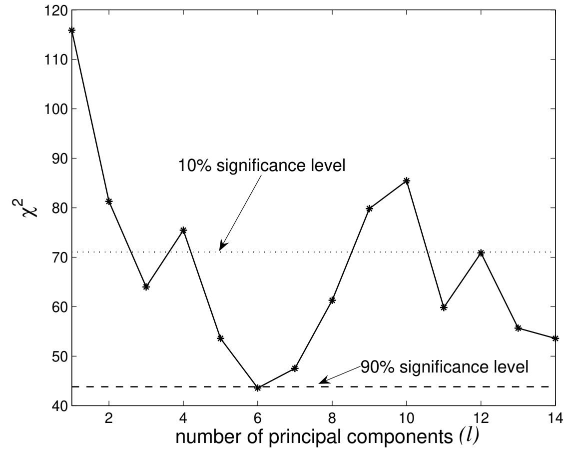


Fig. 3. The value of χ^2 as a function of the number l of leading components for $X_9(t)$ analyzed with $L = 14$.

series. As it can be seen, only the first few principal components are responsible for the main part of the traffic information, that is, the part that maintains a high energy content.

For simplicity, we are interested in grouping the principal components in two subsets, namely, a subset $\mathcal{I}_1$, containing the eigenvalues which are responsible for the reconstruction of a slow varying (trend) component of the original time-series, and a subset $\mathcal{I}_2$, which is related to its "noisy" part. In the standard SSA approach, this partition is performed based on a signal processing point of view. In particular, the subset $\mathcal{I}_1$ consists of the eigenvalues λ_j whose corresponding eigenvectors U_j have slow varying sequences of elements, that is, the contribution of harmonics with *low frequencies* into their Fourier expansion is high. Similarly, the subset $\mathcal{I}_2$ consists of those eigenvalues, for which the contribution of harmonics with *high frequencies* into the Fourier expansion of their corresponding eigenvectors is high. In both cases, the contribution can be measured using the periodogram (16) of each eigenvector.

In our study, instead of following this procedure, the partition of the eigenvalues is based on a statistical criterion, in order to take into account the uncertainty of the underlying statistical model. In particular, the subset $\mathcal{I}_1$ of principal components will consist of those eigenvalues for which the reconstructed time-series has a statistical distribution of the traffic measurements similar to the distribution of the original time-series. Let $p(x)$ denote the probability density function (PDF), which best fits the traffic load series of a given AP. Then, a leading component belongs to $\mathcal{I}_1$, if the PDF of the corresponding reconstructed series, $\hat{p}(x)$, is close to $p(x)$, where the "closeness" is measured using the χ^2 test (17). In this case, the null hypothesis tested by the χ^2 test is that the distribution of the series which is reconstructed using the first l $(1 \leq l \leq L)$ principal components, is modelled with $p(x)$.

As an illustration, Fig. 3 shows the value of the χ^2 test as a function of the number l of the first principal components, for the series $X_9(t)$, whose distribution is best fitted by a Weibull PDF, as it will be described in more detail in Section 4. We used a window size ($L = 14$), which is larger than the "optimal" ($L = 7$), in order to better visualize the variability of the χ^2 test. The two parallel lines correspond to the significance levels $\alpha = 0.1$ (top line) and $\alpha = 0.9$ (bottom line). The significance level indicates the probability that the estimated χ^2 value will exceed the theoretical χ^2 value by chance even for a correct model. For instance, the distribution of the reconstructed series does not pass the null hypothesis (that is, it cannot be modeled as a Weibull PDF), when the reconstruction is based only on the first leading component, since the corresponding χ^2 value is out of the confidence interval (parallel lines). On the other hand, as l increases the value of χ^2 decreases, and for $l = 5$, there is already a quite good level of correspondence of the distribution of the reconstructed series, using the first 5 leading components, to the null hypothesis. This is important, since only the first 5 components contain the main part of the original time-series. Notice that the statistical criterion is consistent with an energy-based rule, i.e., the first five components satisfying the χ^2 test, also contain a high portion of the total energy of the original series ($\approx 80\% = \sum_{i=1}^{5} R_i$).

3.4. *Analysis of residual components*

As it was mentioned before, the influence of the residual components, corresponding to the smallest eigenvalues of the trajectory matrix is related to small irregular variations that do not fit in the basic model of the traffic load and can be interpreted as stochastic noise. As an illustration, Fig. 4 presents the time-series reconstructed on the basis of the smallest residual component of the original series $X_9(t)$, using the window length ($L = 7$) given by the sample auto-correlation function. This time-series has a significantly different structure compared to its original version. Its distribution approximates a Gaussian density, as

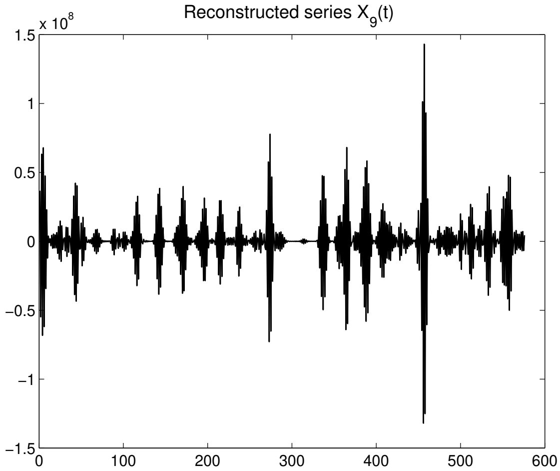


Fig. 4. The series $X_9(t)$ reconstructed using the smallest residual component, for a window length $L = 7$.

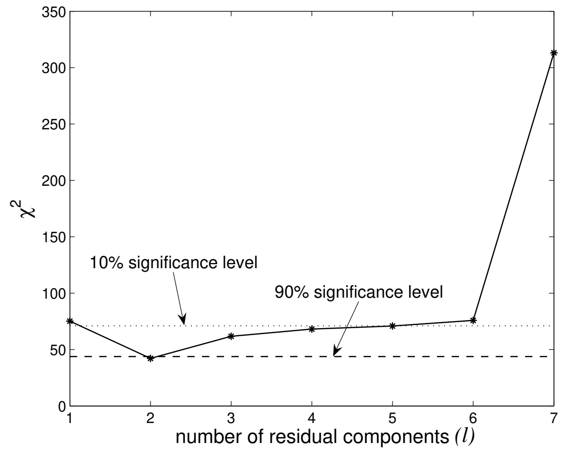


Fig. 6. The value of χ^2 as a function of the number l of residual components for $X_9(t)$ analyzed with $L = 7$.

the Quantile-Quantile (qq)-plot test in Fig. 5 shows. Notice that this reconstructed series does not represent physical traffic load, since it takes negative values.

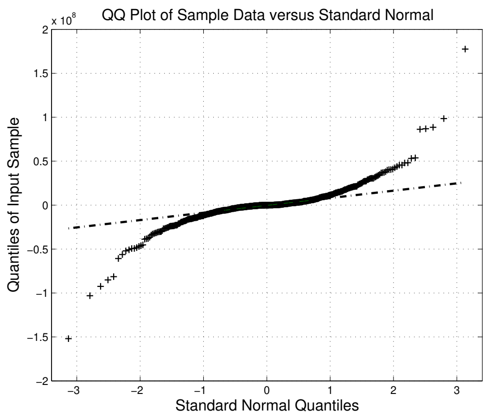


Fig. 5. QQ-plot of the reconstructed series $X_9(t)$ using the smallest residual component, for a window length $L = 7$.

However, when increasing the number of residual components, their distribution starts losing its initial form, together with an increase of the correlations between the series samples. Fig. 6 shows the value of the χ^2 test as a function of the number of *residual* components l ($1 \leq l \leq L$), that is, the first l residual components corresponding to the first l *smallest* eigenvalues, together with the two significance levels ($\alpha = 0.1$–top line, $\alpha = 0.9$–bottom line).

To select the residual components (elements of the $\mathcal{I}_2$ subset), we employ the instance where the symmetry of the distribution of the reconstructed time-series using the first l residual components (smallest eigenvalues) is violated. It can be seen that χ^2 exceeds the 10% ($\alpha = 0.1$) significance level when the number of residual components is equal to $l = 6$. Thus, the subset $\mathcal{I}_2$ contains the first five smallest (non-zero) eigenvalues ($l - 1 = 5$).

Following the analysis described in the previous sections, we constructed two subsets of eigenvalues:

(i) subset $\mathcal{I}_1$ containing the leading components, and

(ii) subset $\mathcal{I}_2$ containing the residual components of the trajectory matrix $\mathbf{H}$ for a given time-series $X(t)$.

Notice that in general $\mathcal{I}_1 \cup \mathcal{I}_2 \subseteq \mathcal{I}$, that is, the analysis of the leading and residual components may not result in an exact partitioning of the initial set of eigenvalues, $\mathcal{I}$, since there are eigenvalues that may not belong in any of the two subsets. Thus, we should replace the equation in (3) (for $s = 2$) by an approximation

$$\mathbf{H} \approx \mathbf{H}^{\mathcal{I}_1} + \mathbf{H}^{\mathcal{I}_2} \; . \tag{7}$$

The application of diagonal averaging (inverse Hankelization) on both sides of (7) results in the following approximation of the original time-series:

$$X(t) \approx X^{\mathcal{I}_1}(t) + X^{\mathcal{I}_2}(t) \tag{8}$$

where $X^{\mathcal{I}_1}(t)$ is the time-series reconstructed using the subset of leading components, which can be interpreted as the trend of $X(t)$, and $X^{\mathcal{I}_2}(t)$ is the time-series reconstructed using the subset of residual components, which can be interpreted as the "noisy" (high-frequency) part of $X(t)$.

4. SSA-based traffic load series modelling and decomposition

As discussed in Section 3, we aim at applying the SSA to decompose the traffic load series of a given AP into its constituent set of eigenloads. Besides, Section 3.3 maintained that the determination of the most important (leading) eigenloads is based on the selection of a suitable statistical model, which accurately fits the distribution of the original traffic load series, to be used in the χ^2 test. Thus, a statistical analysis for the selection of the best model is necessary.

This section first shows that traffic load series corresponding to the total amount of bytes received (download) and sent (upload), as well as, the aggregate amount of traffic, from all clients that are associated with a particular AP can be modelled using PDFs belonging to different families. Then, we use the corresponding best models to show that only a small set of eigenloads can reconstruct the original time-series accurately, while preserving its characteristic features, such as its spikes.

4.1. *Statistical modelling of traffic load series*

The first step in our statistical analysis is based on accurate modelling of the mode and tails of the distribution of a given traffic load series. Since the time-series in our dataset are in general bursty, we expect that their distributions will be modelled using non-Gaussian PDFs. Before proceeding, we assess whether the data deviate from the normal distribution using qq-plots. Then, we determine the model that best fits the empirical distribution of the time-series by employing the so-called amplitude probability density (APD) curves, which represent the probability $P(|X| > x)$. The APD curves give a good indication of whether or not a particular model matches our data near the mode and on the tails of the empirical distribution.

Table 1 indicates the candidate statistical models used in our analysis. The statistical fitting of each one of the 19 time-series constituting our dataset showed that the Gamma is the dominant distribution followed by the GGD. Fig. 7 shows the APD curves for the aggregate traffic load series $X_2(t)$, as well as, the upload traffic and the download traffic, whose empirical distribution is best approximated using the GGD, Gamma and Weibull distribution, respectively.

4.2. *Normalization of the traffic measurements*

Due to the nature of wireless traffic, the traffic load of a particular AP i within hour t, $X_i(t)$, exhibits spikes that are very hard to predict. Fig. 8 shows the traffic load series and its normal probability plot for the third hotspot AP in our dataset, $X_3(t)$. Its bursty behavior is clear, and the marginal distribution is non-Gaussian. Thus, before proceeding to handle the data more efficiently, we normalize them so that their distribution resembles the normal distribution, by employing a suitable transformation. Normal distributions can be handled more effectively. Besides, such a transformation can reduce the effect of those high local spikes on the forecasting performance. Unfortunately, in general, the choice of the best transformation is not obvious.

There is a family of power transformations to make the marginal distributions resemble a Gaussian-like density, namely the Box-Cox power transformations, defined only for positive data values. The existence of zero values in the trace does not pose any problem, since a constant can always be added. The Box-Cox power transformation is given by (18):

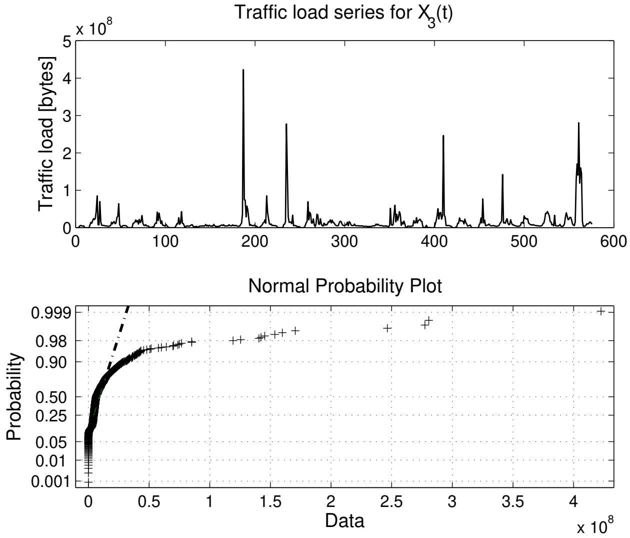


Fig. 8. Traffic load series and normal probability plot for $X_3(t)$.

$$x(\rho) = \begin{cases} \dfrac{x^\rho - 1}{\rho}, & \rho \neq 0 \\ \ln(x), & \rho = 0 \end{cases} \tag{9}$$

Given the time-series $X = \{x_j\}_{j=1}^N$, one way to select the optimal power ρ is to maximize the log-likelihood function

$$f(X, \rho) = -\frac{N}{2} \ln\Big[\sum_{j=1}^{N} \frac{(x_j(\rho) - \bar{x}(\rho))^2}{N}\Big] + (\rho - 1) \sum_{j=1}^{N} \ln(x_j(\rho)) \ , \tag{10}$$

where

$$\bar{x}(\rho) = \frac{1}{N} \sum_{j=1}^{N} x_j(\rho) \ .$$

Let $Y_i(t) = X_i(t; \rho_{opt})$ denote the transformed version of the original time-series $X_i(t)$ using the optimal value of ρ (ρ_{opt}). Fig. 9 shows the Box-Cox transformed version ($Y_3(t)$) of the original series $X_3(t)$, as well as, the mean and standard deviation per hour-of-day ($h(t) \in \{1, \ldots, 24\}$) of the original series $X_3(t)$. It is apparent that the effect of the large spikes, present in $X_3(t)$, has been degraded in $Y_3(t)$. Besides, notice that $X_3(t)$ exhibits strong non-stationarity in both the mean and the standard deviation (see the two bottom plots in Fig. 9).

This analysis motivates us to further normalize the transformed time-series, $Y_i(t)$, in the following way:

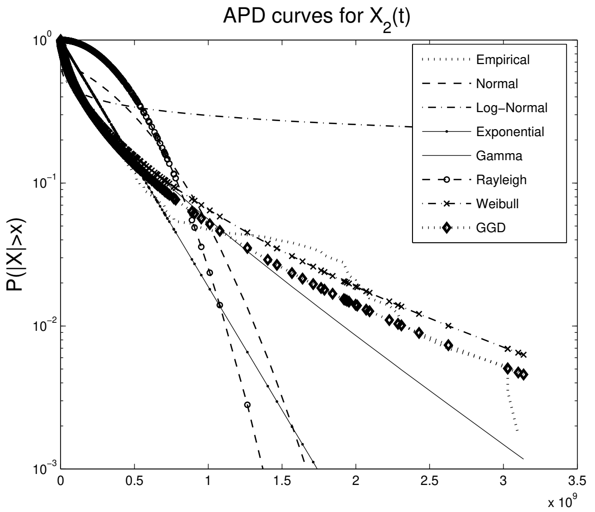
$$G_i(t) = \frac{Y_i(t) - \mu_{i,h(t)}}{\sigma_{i,h(t)}} \ , \tag{11}$$

where $h(t)$ is the corresponding hour-of-day for time t, while $\mu_{i,h(t)}$ and $\sigma_{i,h(t)}$ are the mean and standard deviation of $Y_i(t)$ during those time periods with the hour-of-day being $h(t)$, respectively.

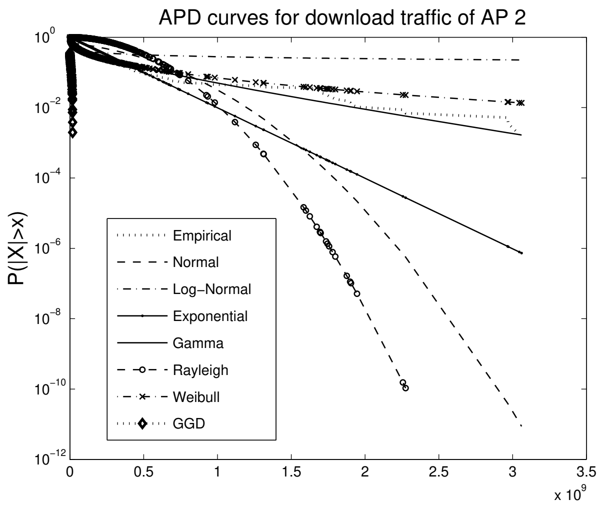
Notice that the computation of $\mu_{i,h(t)}$ and $\sigma_{i,h(t)}$ depends on the periodicity of the measurements of the particular AP. In particular, when the AP has a diurnal periodicity (24

Table 1
The models used in the traffic load series analysis.

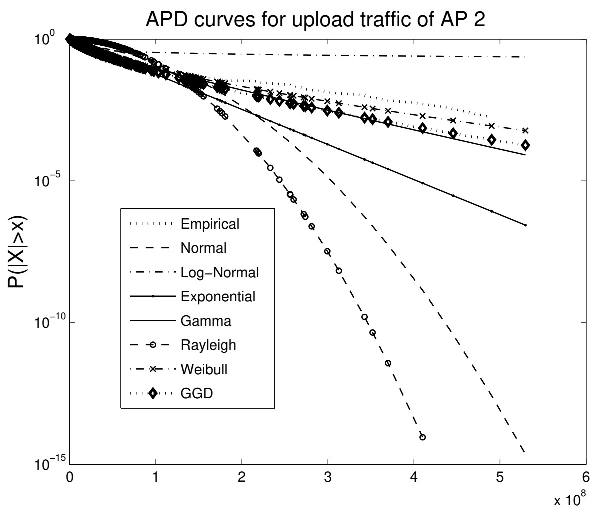
Model	PDF
Normal	$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/(2\sigma^2)}$
log-Normal	$p(x) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-(\ln x-\mu)^2/(2\sigma^2)},\ x \in (0,\infty)$
Exponential	$p(x) = \frac{1}{\mu} e^{-x/\mu}$
Gamma	$p(x) = \frac{1}{b^\alpha\,\Gamma(\alpha)} x^{\alpha-1} e^{-x/b},\ x \in [0,\infty)$
Rayleigh	$p(x) = \frac{x}{b^2} e^{-x^2/(2b^2)},\ x \in [0,\infty)$
Weibull	$p(x) = b\alpha^{-b} x^{b-1} e^{-(x/\alpha)^b},\ x \in [0,\infty)$
Generalized Gaussian density (GGD)	$p(x) = \frac{b}{2\alpha\,\Gamma(1/b)} e^{-(\lvert x\rvert/\alpha)^b}$



(a) APD curves of the time-series $X_2(t)$



(b) APD curves of the download time-series



(c) APD curves of the upload time-series

Fig. 7. APD curves for the time-series corresponding to the traffic of AP 2: (a) aggregate traffic, (b) download trafic, (c) upload traffic.

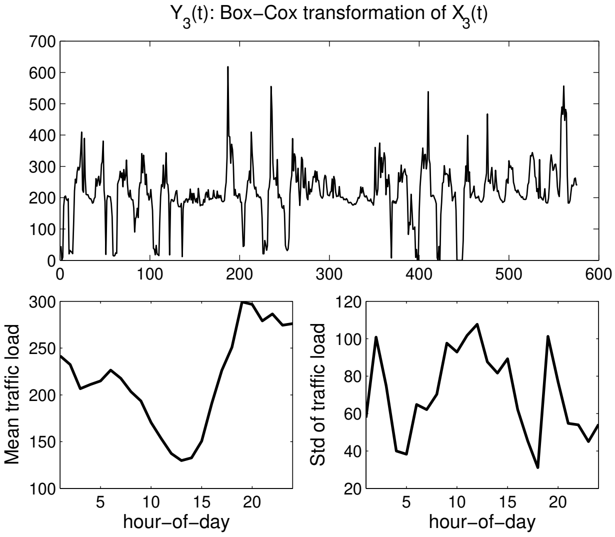


Fig. 9. Box-Cox transformation of $X_3(t)$ (upper figure) and the patterns of the mean and standard deviation per hour-of-day of the original traffic load.

hours), $\mu_{i,h(t)}$ is the mean of the traffic measurements obtained at the same hour-of-day $h(t)$. Assume, for instance, that we want to normalize the measurements obtained at 11:00 a.m. for each day. Then, $\mu_{i,h(t)}$ is the mean of all the traffic measurements corresponding only to 11:00 a.m., where the mean is taken over all the days in our trace. On the other hand, we must be careful when the periodicity of an AP is not diurnal. For instance, assume that we want to normalize the measurements of an AP with a periodicity of 22 hours, obtained at 11:00 a.m. for each day. Then, $\mu_{i,h(t)}$ is the mean of all the traffic measurements corresponding to hours-of-day, whose difference is equal to 22 hours between adjacent days. The computation of the standard deviation, $\sigma_{i,h(t)}$, is performed in the same way.

It is also important to notice that after the above transformations, the model that best fits the empirical distribution of $G_i(t)$ and the optimal window length L, given by the sample auto-correlation function, may change.

4.3. *Low-dimensionality of traffic load series*

As described in Section 3.3, the energy contributed by each eigenload to the actual traffic load is concentrated in the first few leading components. Fig. 10 shows the percentage contribution of the eigenvalues for the normalized series $G_9(t)$, for the optimal window length $L = 33$, as well as, the χ^2 test, where the null hypothesis is that the Weibull PDF is closer to the empirical distribution. According to the χ^2 test, only the first six principal components (out of the 33) are adequate to capture the vast majority of traffic variability, and thus, accurately approximate the original series $X_9(t)$. The first six principal components contribute in the 72.42% of the total energy of $G_9(t)$. Thus, the time-series $G_9(t)$ has a structure with effective dimension equal to six, much lower than the total number of principal components (33). The χ^2 test is consistent with an energy-based rule, in the sense that, the few first principal components for which the χ^2 value falls inside the confidence interval are exactly those components containing the highest portion of the total energy.

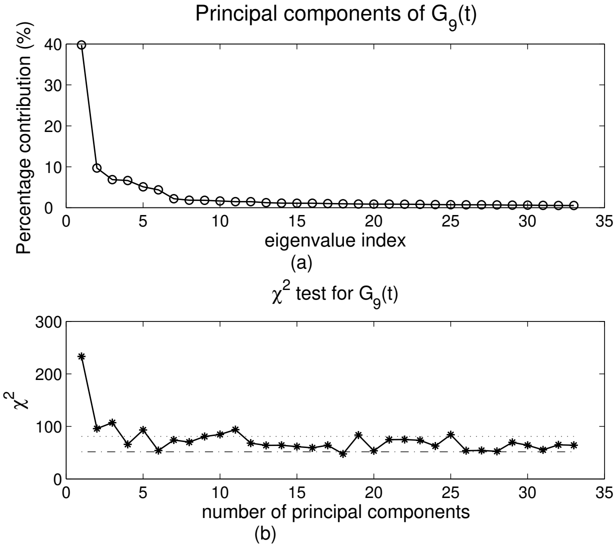


Fig. 10. (a) Percentage contribution of the eigenvalues and (b) χ^2 test values for $G_9(t)$ ($L = 33$).

As a further illustration of this low-dimensionality property of the traffic load series, we plot the time-series $\hat{G}_9(t)$, which is reconstructed using the first 6 leading eigenloads. The results are shown in Fig. 11(a), while Fig. 11(b) presents the original series of the aggregate traffic load $X_9(t)$, and its approximation $\hat{X}_9(t)$, obtained after applying the inverse transforms of Eqs. (9), (11) (in this order). Notice that even though we omitted 27 principal components, we can still capture most of the important characteristics of the original series $X_9(t)$, such as the locations of its spikes. Fig. 12 shows the approximation results using the set of leading components for the normalized upload and download traffic-series of the 9-th AP. In particular, Fig. 12(a) illustrates the normalized upload traffic-series, analyzed with a window size $L = 35$, along with its approximation using the first 13 leading components, while in Fig. 12(b), we plot the normalized download traffic-series, analyzed with a window size $L = 32$, along with its approximation using the first 5 leading components. From the above figures it is clear that the SSA approach is suitable for the approximation not only of the aggregate traffic load series, but also of its two additive components, namely the upload and download traffic. Notice that in general these three time-series (upload, download, aggregate) are analyzed using different window sizes, L, and statistical distributions.

We do not expect to capture accurately the exact height of a spike, using only such a small number of eigenloads. Our goal is to illustrate that the main information content of a traffic load series in our dataset is mainly due to the contribution of a small number of features (eigenloads). Thus, we can understand better the intrinsic behavior of actual traffic by studying the behavior of a small set of eigenloads, which appear to have a better structure compared with the original traffic load series, as described in the next section.

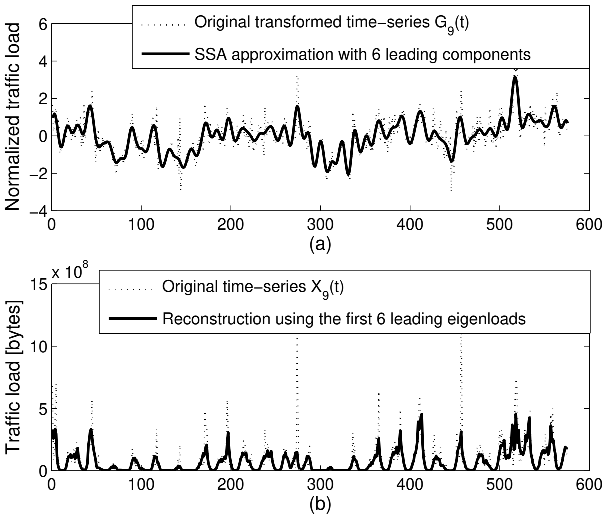


Fig. 11. SSA approximation using the first 6 leading components for: (a) the normalized series $G_9(t)$, (b) the original series $X_9(t)$.

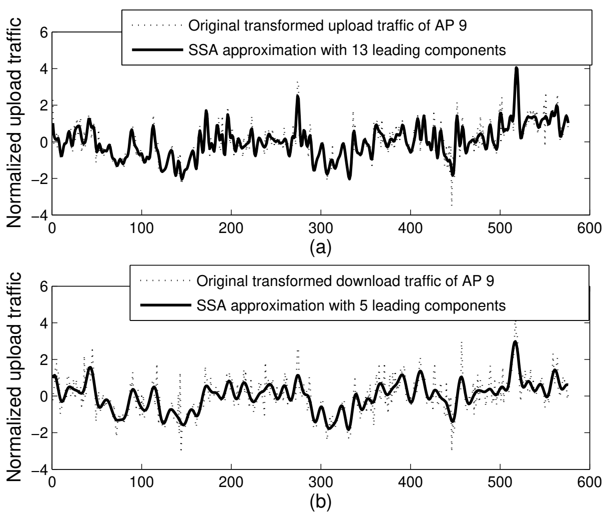


Fig. 12. SSA approximation using the first K leading components for: (a) the normalized upload series ($K = 13$), (b) the normalized download series ($K = 5$), of the 9-th AP.

5. Structure of the eigenloads

The statistical analysis of the traffic load series presented in the previous section underscores the central role of the eigenloads in understanding the intrinsic properties of a traffic load series obtained in a large-scale WLAN, such as the network considered in our study. Thus, we are interested in describing the two types of eigenloads, namely, the deterministic (slow-varying) and the noisy one.

5.1. *Categorization of eigenloads*

In Section 3.1, we defined the set of eigenloads $\{V_j\}_{j=1}^r$ as a function of the set of eigenpairs $\{\lambda_j,\, U_j\}_{j=1}^r$. The value of λ_j is proportional to the extend to which its corresponding principal component U_j contributes to the j-th eigenload of the time-series $X(t)$. Thus, before the categorization of the eigenloads, we start by inspecting the set of principal components $\{U_j\}_{j=1}^r$.

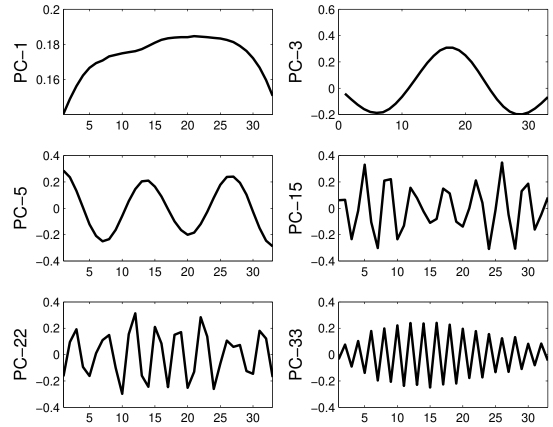
The principal components of the traffic load series (upload, download, and aggregate) in our dataset for each one of the 19 hotspot APs appear to have the same behavior. In particular, we found that a principal component whose corresponding eigenvalue is of high magnitude is slow-varying, whereas as the magnitude of an eigenvalue decreases, its corresponding eigenvector oscillates more and more. As an illustration of this behavior, Fig. 13 shows a subset of the principal components for the normalized aggregate traffic-series $G_9(t)$, as well as, its corresponding upload and download traffic-series, where the principal components are ordered in decreasing order with respect to their corresponding eigenvalue.

The categorization of the set of eigenloads is performed in a heuristic way. In particular, we expect that the eigenloads will present a similar behavior as the principal components, since they are obtained as projections of the trajectory matrix $\mathbf{H}$ on them. Thus, we divide the eigenloads in two classes: (i) the deterministic, slow-varying eigenloads, which are simply the projections of $\mathbf{H}$ on slow-varying principal components, and (ii) the noisy eigenloads, which result by projecting $\mathbf{H}$ on the high-frequency principal components. As an example, Fig. 14 shows the eigenloads given by the projection of the trajectory matrix of the normalized time-series $G_9(t)$ on the principal components 1, 5, 22 and 33. It is clear that the eigenloads 1 and 5 can be considered as deterministic, while the eigenloads 22 and 33 belong to the noisy class.

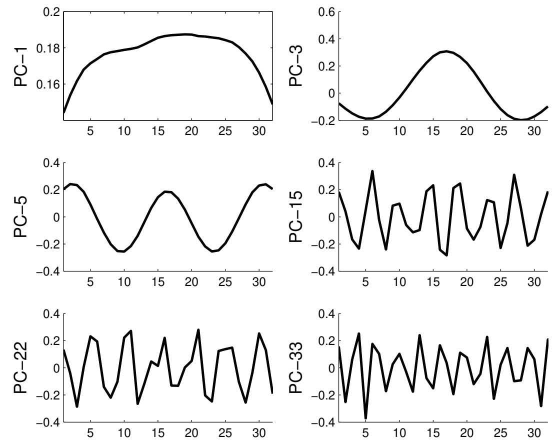
5.2. *Decomposition of the traffic load series*

A benefit of the above eigenload-based categorization is that it results in a decomposition of any given traffic load series (normalized or not) into its principal features. That is, we can reconstruct each time-series in terms of two constituents: the contributions made by the deterministic and the noisy eigenloads. Doing so, each constituent is responsible to capture a distinct feature of the traffic load series, namely, its deterministic mean and its (stationary) random variation, respectively. An example of this decomposition is shown in Fig. 15. The figure shows the original normalized traffic series $G_9(t)$ along with its four approximations based on the features captured by the first four eigenloads (i.e., the eigenloads corresponding to the first four leading components).

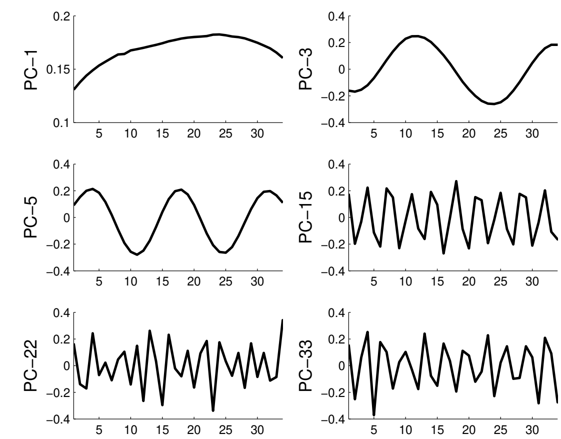
This decomposition may be very useful for the future design of a forecasting system, since the eigenloads which



(a)



(b)



(c)

Fig. 13. Principal components of the normalized traffic-series: (a) aggregate $G_9(t)$, (b) upload, (c) download.

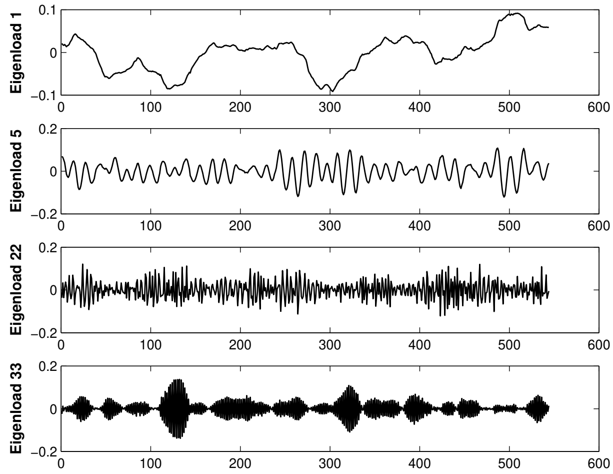


Fig. 14. Eigenloads of the traffic load series $G_9(t)$.

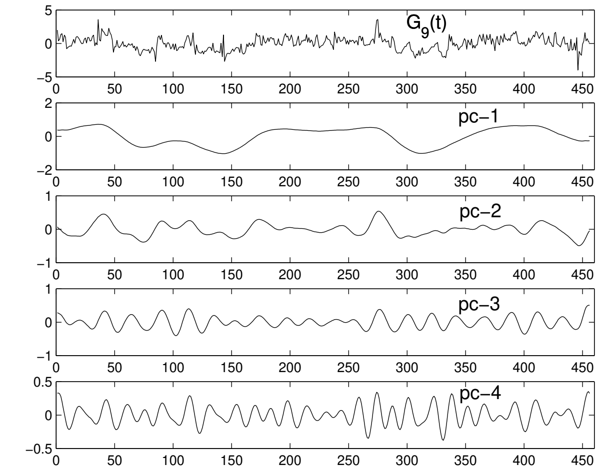


Fig. 15. Original normalized traffic load series $G_9(t)$ along with its approximations based on the first four eigenloads.

carry most of the information content of the original time-series are exactly the first few deterministic ones, which can be predicted more accurately because of their slow-varying behavior.

6. Traffic Forecasting

In the following, we exploit the multi-scale decomposition provided by the SSA in the development of a forecasting algorithm. In particular, we focus on the prediction of traffic values in a relatively long-time horizon, based on the estimated deterministic component (trend), which captures the low-frequency behavior of the original time-series.

An initial attempt toward long-term forecasting of IP network traffic is described in (19), where a single value for the aggregate number of bytes flowing over the NSFNET is computed, and linear time-series models are used for modelling the traffic behavior. Specifically, the time-series obtained are modelled with a low-order autoregressive integrated moving average (ARIMA) model, offering highly accurate forecasts for up to two years in the future. However, predicting a single value for the future network-wide load could be relevant for dynamic resource allocation in small time-scales (20; 21), but it is insufficient for capacity planning purposes. In this case, it is necessary to develop models in larger time-scales for long-term forecasting. An initial attempt to model the evolution of IP backbone traffic at large time-scales is described in (22), which relies on the wavelet multiresolution analysis and linear time-series models. In particular, the collected measurements are smoothed using a wavelet decomposition in multiple scales, until the overall long-term trend is identified. Then, the forecasting of this trend component is achieved by fitting it with a low-order ARIMA model.

In our work, we are interested in predicting the behavior of the trend component of the traffic load in a campus-wide WLAN, at a larger than an hourly time-scale, for instance to predict the trend in the next few days. Our measurements come from a dynamic environment reflecting events that may have short or long-time effects on the observed behavior. The long-time events influence the overall long-term trend, and the main part of the variability observed. Events that may have a short-time duration, such as link failures, usually have a direct impact on the measured traffic but their effect fades after some time. As a consequence, they contribute to the measured traffic-series with values which lie beyond the overall trend. Given that such events are very hard to predict, we will not attempt to model them in this paper, but we provide some directions for future research in the last section.

6.1. *Traffic predictor overview*

As mentioned before, the trend component of the original (normalized) traffic-series is obtained by projecting it on the set of leading components, given by the SSA. In the following, let $G_i^d(t)$ denote the deterministic component (trend), of the original normalized traffic-series $G_i(t)$. The design of a predictor for $G_i^d(t)$ is based on its fitting with a p-th order linear model of the form

$$\hat{G}_i^d(t+1) = \sum_{l=0}^{p-1} w_t(l) G_i^d(t-l) \ , \tag{12}$$

where $\mathbf{w}_t = [w_t(0), w_t(1), \ldots, w_t(p-1)]^T$ denotes the predictor's weight vector at time t. Besides, let $e(t) = G_i^d(t+1) - \hat{G}_i^d(t+1)$ denote the prediction error. The Normalized Least Mean Squares (NLMS) (23) algorithm uses error feedback to update successively the weight vector as follows,

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \frac{\mu\, e(t-1)\mathbf{G}_i^d(t-1)}{\|\mathbf{G}_i^d(t-1)\|^2} \ , \tag{13}$$

Table 2
Optimal order p for the BRECV, BSENT and aggregate time-series of a subset of 10 AP's.

AP index	1	2	4	6	8
BRECV	21	32	22	22	12
BSENT	32	80	32	30	32
Aggregate	30	10	12	40	70
AP index	10	11	14	16	18
BRECV	12	12	22	32	32
BSENT	32	32	12	32	70
Aggregate	12	12	12	42	80

where $\mathbf{G}_i^d(t-1) = [G_i^d(t-1), G_i^d(t-2), \ldots, G_i^d(t-p)]^T$ and μ is a step size parameter. Notice that the error $e(t-1)$ is used instead of $e(t)$ in (13), since $e(t)$ is not available at time t (it requires the knowledge of the future value $G_i^d(t+1)$). Besides, the selection of μ such that $0 < \mu < 2$ is necessary for the convergence of the weight vector (23). In our proposed method, we set $\mu = 0.5$ and the weights of the NLMS predictor are adapted periodically, after new measurements become available in a predefined time interval.

The selection of the model's order p, is equivalent to selecting an appropriate window size in terms of the number of samples of past traffic load, and controls the amount of traffic history measurements to be used in forecasting the future traffic load. In the proposed method, we select the value of p by fitting a family of models to a subset of $G_i^d(t)$ values (e.g. the traffic load in the first 5 days), for several values of p, and select the optimal value of p by minimizing the Akaike Information Criterion (AIC) (24). Although in our method this optimal value is maintained during the whole forecasting period, one could also re-estimate the optimal p in certain periods. Table 2 shows the optimal order for the time-series containing the amount of bytes received (BRECV), sent (BSENT), as well as, the aggregate traffic load, from all clients that were associated with a particular AP, for a subset of 10 out of the 19 hotspot AP's in our dataset. These 10 AP's are selected as the ones for which the SSA approach resulted in the best statistical modelling using one of the distributions shown in Table 1.

We observe that for the same AP, the optimal order of the two separate traffic load series, namely the BRECV and BSENT, can be very different from each other, which is due to their different variation. Usually, the predictor's order p increases as the non-stationarity of the time-series increases.

The proposed forecasting scheme collects the traffic load measurements at certain time intervals. These values are the inputs for the traffic predictor. After each sampling, a prediction is made for the traffic load in the next sampling period. This forecast could be then used in the future management of the network's capacity, such as in deciding whether to admit or reject new associations in a certain AP.

6.2. *Evaluation of traffic load forecasting*

In the following, we apply the proposed methodology to predict the trend of the traffic series corresponding to each one of the 10 AP's, whose indexes are shown in Table 2. In particular, we evaluate the prediction performance for the BRECV and BSENT time-series, as well as for the time-series containing the aggregate traffic load, which is simply the sum of BRECV and BSENT. These time-series contain the traffic load for a period of 58 days. The NLMS approach is employed to update the weight vector $\mathbf{w}_t$ every two days. That is, the traffic load values in a window, whose size depends on the model's order p, are used to predict the trend in the next two days. Then, the true measurements collected during these two days are used for updating the weight vector and predicting the trend for the next two days and so on.

Two metrics are used for the evaluation of the forecasting performance, namely i) the absolute relative prediction error with respect to the true trend (we denote it by $E1$) and ii) the absolute relative prediction error with respect to the true total traffic load (denoted by $E2$), which are defined as follows,

$$E1(t) = \Big| \frac{G_i^d(t) - \hat{G}_i^d(t)}{G_i^d(t)} \Big| \, , \tag{14}$$

$$E2(t) = \Big| \frac{G_i(t) - \hat{G}_i^d(t)}{G_i(t)} \Big| \, . \tag{15}$$

Fig. 16 shows the median relative prediction error $E1$ for the time-series of the upload (BSENT), the download (BRECV), as well as, the aggregate traffic load, for each one of the selected 10 AP's, while Fig. 17 shows the corresponding median relative prediction errors $E2$. From the first figure we can see that the trend component can be predicted with high accuracy for these AP's, while the second one shows that for most of the 10 AP's, the predicted value of the trend is relatively close to the total traffic load, which is also an indication that the trend component carries the major part of the information content of the original time-series. We expect that the forecasting performance can be improved by taking into account the residual components, which constitute the noisy part of the original time-series and are responsible for the high-frequency fluctuations.

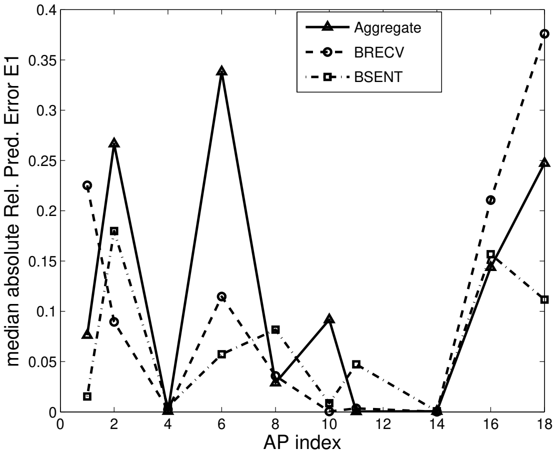


Fig. 16. Median absolute relative prediction error $E1$.

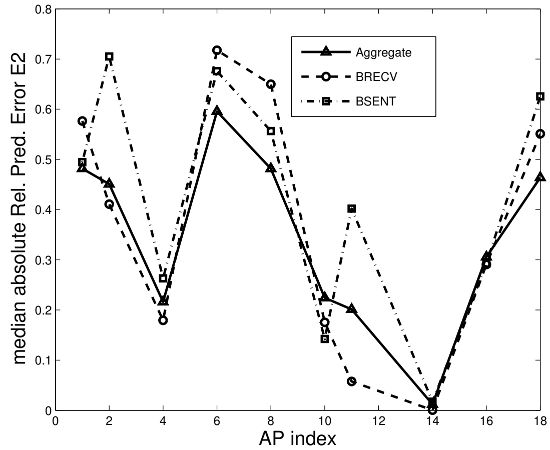


Fig. 17. Median absolute relative prediction error $E2$.

7. Conclusions

In this paper, we provided an SSA-based statistical analysis of the structure of traffic load series measured in a large-scale campus-wide WLAN. First, we fitted the distribution of a given time-series containing the upload, download or the aggregate traffic load, obtained at the AP level, using an appropriate model selected from a set of pre-determined candidate PDFs. Then, we applied the SSA approach in order to partition the set of principal components in two subsets, namely, the subset of leading components and the subset of residual components.

We showed that the subset of leading components is responsible for the preservation of the main information content of the original series and thus, the intrinsic dimensionality of the traffic is highly restricted by using only the first few leading components. Besides, we found that the eigenloads, defined using the set of leading components, present a similar behavior across the different traffic load series. In particular, we showed that they can be categorized in two classes: (i) the deterministic, slow-varying eigenloads carrying the major information content and (ii) the noisy eigenloads, which are related with the irregular variations of the traffic.

Based on this categorization, we decomposed the original traffic series by projecting it on each eigenload. This yielded a considerable understanding of the structure of the traffic load series, which is analyzed in multiple frequency scales, since the projections on the first eigenloads give the slow-varying trend components, while the projections on the last eigenloads give the high-frequency content.

Motivated by the common behavior of the eigenloads among the several AP's, we exploited the slow-varying trend components to design a traffic predictor. The results revealed an increased forecasting performance of the deterministic part, for the traffic-series of those AP's for which the SSA approach provided an accurate statistical modelling.

As a future work, we will complete the design of the proposed predictor by taking into account not only the deterministic, but also the noisy component of a given traffic-series. For this purpose, an optimal radial basis function will be trained for the prediction of the noisy part (25).

Another interesting problem is the detection of dynamical changes of the future traffic load values. In particular, the accurate detection of transitions from a normal to an abnormal state, either due to hardware or software failure, or due to an attack, may improve diagnosis and treatment. The multi-scale decomposition given by the SSA approach, could be combined with the conceptually simple and computationally very fast concept of permutation entropy (26) to detect dynamical changes in the subset of noisy eigenloads, which are responsible for the transient behavior of the traffic load series. Besides, to encourage further experimentation, we have made our datasets available to the research community (32).

References

[1] T. Henderson, D. Kotz, and I. Abyzov, "The changing usage of a mature campuswide wireless network", *In ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Philadelphia, Sep. 2004.

[2] M. Ploumidis, M. Papadopouli, and T. Karagiannis, "Multi-level application-based traffic characterization in a large-scale wireless network", *in Proc. of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Helsinki, Finland, June 2007.

[3] F. Anjum, M. Elaoud, D. Famolari, A. Ghosh, R. Vaidyanathan, A. Dutta, P. Agrawa, T. Kodama, and Y. Katsube, "Voice performance in WLan networks, an experimental study", *in Proc. of the IEEE Conference on Global Communications (GLOBECOM)*, Rio De Janeiro, Brazil, Dec. 2003.

[4] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of ethernet traffic", *ACM Computer Communication Review*, 25(1):202–213, 1995.

[5] W. E. Leland, W. Willinger, M. S. Taqqu, and D. V. Wilson, "Statistical analysis and stochastic modeling of self-similar datatraffic", *in Proc. 14th Int. Teletraffic Cong.*, Vol. 1, pp 319-328, Antibes Juan Les Pins, France, June 1994.

[6] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. .D. Kolaczyk, and N. Taft, "Structural Analysis of Network Traffic Flows", *ACM Sigmetrics*, New York, June 2004.

[7] F. H. Campos, M. Karaliopoulos, M. Papadopouli, and H. Shen, "Spatio-Temporal Modeling of Traffic Workload in a Campus WLAN", *2nd annual intl. WIreless internet CONference (WICON'06)*, Boston, USA, August 2-5, 2006.

[8] M. Karaliopoulos, M. Papadopouli, E. Raftopoulos, and H. Shen, "On scalable measurement-driven modelling of traffic demand in large WLans", *in Proc. of the IEEE Workshop on Local and Metropolitan Area Networks*, Princeton NJ, USA, June 10-13, 2007.

[9] M. Papadopouli, H. Shen, E. Raftopoulos, M. Ploumidis, and F. Hernandez-Campos, "Short-term traffic forecasting in a campus-wide wireless network", *16th Annual IEEE Intl. Symp. on Personal Indoor and Mobile Radio Comm.*, Berlin, Germany, September 11-14, 2005.

[10] M. Papadopouli, E. Raftopoulos, and H. Shen, "Evaluation of short-term traffic forecasting algorithms in wireless networks", *2nd Conf. on Next Generation Internet Design and Engineering*, Valencia, Spain, April 3-5, 2006.

[11] America's most connected campuses. `http://forbes.com/home/lists/2004/10/20/04conncampland.html`.

[12] H. D. I. Abarbanel, *"Analysis of Observed Chaotic Data"*, Springer-Verlag New York, Inc., 1996.

[13] I. T. Jolliffe, *"Principal Component Analysis"*, Springer-Verlag, 1986.

[14] N. Golyandina, V. Nekrutkin, and A. Zhigljavsky, *"Analysis of Time Series Structure: SSA and Related Techniques"*, Chapman & Hall/CRC, 2001.

[15] I. Antoniou, V. .V. Ivanov, Valery V. Ivanov, and P. V. Zrelov, "Principal Component Analysis of Network Traffic Measurements: the "Caterpillar"-SSA approach", *"VIII Int. Workshop on Advanced Computing and Analysis Techniques in Physics Research, ACAT'2002*, 24-28 June 2002, Moscow Russia.

[16] M. H. Hayes, *"Statistical Digital Signal Processing and Modeling"*, John Wiley & Sons, 1996.

[17] P. E. Greenwood, and M. S. Nikulin, *"A Guide to Chi-Squared Testing"*, John Wiley & Sons Canada, Ltd., 1996.

[18] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *"Time Series Analysis, Forecasting and Control"*, 3rd ed. Prentice Hall, Englewood Clifs, NJ, 1994.

[19] N. K. Groschwitz, and G. C. Polyzos, *"A Time Series Model of Long-Term NSFNET Backbone Traffic"*, in *IEEE ICC94*, 1994.

[20] S. Basu, and A. Mukherjee, *"Time Series Models for Internet Traffic"*, in *24th Conf. on Local Computer Networks*, Oct. 1999, pp. 164–171.

[21] A. Sang, and S. Li, *"A Predictability Analysis of Network Traffic"*, in *INFOCOM*, Tel Aviv, Israel, Mar. 2000.

[22] K. Papagiannaki, N. Taft, Z. -L. Zhang, and C. Diot,

"Long-Term Forecasting of Internet Backbone Traffic", in *IEEE Trans. on Neural Networks*, 16(5):1110–1124, Sep. 2005.

[23] M. H. Hayes, *Statistical digital signal processing and modeling*, John Wiley & Sons, Inc., New York, 1996.

[24] P. F. Brockwell, and R. A. Davis, *"Time Series: Theory and Methods"*, New York: Springer-Verlag, New York, 1998.

[25] H. Leung, T. Lo, and S. Wang, *"Prediction of Noisy Chaotic Time Series Using an Optimal Radial Basis Function Neaural Network"*, in *IEEE Trans. on Neural Networks*, 12(5):1163–1172, Sep. 2001.

[26] Y. Cao, W. W. Tung, J. B. Gao, V. A. Protopopescu, and L. M. Hively, *"Detecting dynamical changes in time series using the permutation entropy"*, in *Physical Review* E 70, 046217, 2004.

[27] H. Shen and J. Z. Huang, *"Interday Forecasting and Intraday Updating of Call Center Arrivals"*, in *Manufacturing & Service Operations Management (MSOM), Articles in Advance*, pp. 1–20, Jan. 2008.

[28] M. Papadopouli, H. Shen and M. Spanakis, *"Modeling client arrivals at access points in wireless campus-wide networks"*, in *14th IEEE Workshop on Local and Metropolitan Area Networks*, Chania, Crete, Greece, Sep. 2005.

[29] M. Papadopouli, H. Shen and M. Spanakis, *"Characterizing the duration and association patterns of wireless access in a campus"*, in *11th European Wireless Conference*, Nicosia, Cyprus, Apr. 2005.

[30] G. Tzagkarakis, M. Papadopouli and P. Tsakalides, *"Singular Spectrum Analysis of Traffic Workload in a Large-Scale Wireless LAN"*, in *10th ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Chania, Crete, Greece, Oct. 2007.

[31] F. Chinchilla, M. Lindsey and M. Papadopouli, *"Analysis of wireless information locality and association patterns in a campus"*, in *IEEE INFOCOM 2004*, Hong Kong, Mar. 2004.

[32] UNC/FO.R.T.H. archive of wireless traces, models and tools. `http://netserver.ics.forth.gr/datatraces/`