

# A Method Using Time Series Analysis for IEEE 802.11 WLANs Channel Forecasting

Jeandro Bezerra, Rudy Braquehais, Filipe Roberto,  
Jorge Silva, Marcial Fernandez, Thelmo de Araújo and Celestino Junior  
Laboratório de Redes de Comunicação e Segurança  
Universidade Estadual do Ceará,  
Fortaleza, Ceará, Brasil 60740-903  
Email: {jeandro,rudy,filipe,jlcs,marcial,thelmo,celestino}@larces.uece.br

**Abstract**—The growth of wireless network use has greatly increased research demand. Some applications, which are context-aware, must adapt to the environment. So, information on both environment characteristics and the device's hardware are crucial. In this work, a new method called Natural Adaptive Exponential Smoothing (NAES) is proposed. It describes and forecasts, in real time, IEEE 802.11 WLAN networks channel behavior. The NAES method is a variation of the exponential smoothing technique to compute the channel quality indicators, namely the Received Signal Strength (RSS) and the link quality. A comparison with the results obtained by the Trigg and Leach (TL) method shows that NAES outperforms TL method.

**Keywords**—WLAN, forecasting, exponential smoothing

## I. INTRODUCTION

The use of WLANs (Wireless Local Area Network) with IEEE 802.11 standard has been intense lately. Public WLANs which provide high transfer rate access with no costs to the general public, have become more and more popular in university campi, airports, hotels, and other public places. The development of wireless technology has increased the use of mobile devices and raised the demand for more sophisticated context-aware applications. The proliferation of wireless-enabled devices has aggravated the competition for the always limited bandwidth in the wireless infrastructure, which may soon make the wireless devices victims of their own success. Because mobile devices have hardware limitations, from battery and memory sizes to low cost requirements, software applications might optimize hardware use to improve autonomy.

An analysis of communication quality can be performed by measuring the RSS and the link quality - both can vary considerably on time and space. Unlike some papers [1], [2], that focus on the forecast traffic systematically based on the user's properties and on environment information, we shall use, on this paper, mathematical models to forecast, in real time, the short time behavior of the wireless communication channels. For instance, with the channel behavior prediction, actions to improve the adaptability and management of computational resources (e.g. memory, battery power) could be taken. Specifically, we have developed a forecast algorithm that takes as input the on-line measurements of RSS and the

link quality to forecast the channel behavior. In our method, the forecast is done in real time.

This work is organized as follows: Section II presents some methods for traffic analysis and prognosis in both wired and wireless networks. The parameters of wireless channels we used and some characteristics of the IEEE 802.11 standard are presented in section III. Section IV explains the simple exponential smoothing method and its variants. The method we implemented is in section V, and the results are in section VI. Finally, we present the conclusion and commentaries on future work in section VII.

## II. RELATED WORK

The idea of using prognoses based on time series is used in many areas of knowledge, especially in Applied Social Sciences, Statistics, and Mathematics. But, it is also used in the area of Networks Communications [3], e.g., to predict possible network traffic congestions. Recently some papers proposed forecasting models to Wireless Networks [2], [1], for many applications, listed in this section.

Kelvin [4] proposes a model for calls admission control and resource reservation in mobile networks, which is based on the forecast of handoffs loads, using the adaptive exponential smoothing method [5] (*Trigg and Leach*) to forecast the amount of broadband resources. That work differs from ours in the parameters chosen for data analysis (bandwidth) and because its method aims admission control.

Another proposal for traffic forecasting in mobile networks is presented by Akinaga et al. [1]. The author uses a modified multiple regression method for time series - via established external variable - and proposes a method of forecasting traffic based on the user's properties and information about the environment. The user properties inform when and where the users are at a given time, the users calls duration, and the impact of external factor (e.g., rain, snow).

Rabelo [6] evaluates the quality of the communication by using statistical control methods and spectral correlation via *wavelets*. This works uses the same channel parameters (RSS, link quality) we used to evaluate our methods. An API (Application Programming Interface) is used to share the collected information over various applications.

Chen and Rappaport [2] analyze a forecasting model that predicts throughput using packet information. They use empirical measurements collected in three hotspots (using IEEE 802.11b). Although they also (like us) use data from real environments, the forecast is not done in real time.

### III. COMMUNICATION QUALITY PARAMETERS

The first phase of this work was to identify which parameters would be measured and evaluated in a WLAN. This section presents the IEEE 802.11b architecture standard and describes the measured parameters. Next, we discuss the use of the chosen parameters.

The wireless interface physical layer communicates through radio frequency (RF) in a similar way as an infrared communication. The focus of this work is based on the RF, through spread spectrum, on ISM (Industrial Medical and Scientific) band. Interfaces that work with Direct Sequence Spread Spectrum (DSSS) were used. They operate in DBPSK (Differential Binary Phase Shift Keying) modulation and DQPSK (Differential Quadrature Phase Shift Keying). Their transmission power is under 1000mW (approximately).

The IEEE 802.11b works in the band of 2.4GHz and uses the DSSS to offer the bandwidth of 1Mbps and 2Mbps. Two new rates - 5.5Mbps and 11Mbps - are obtained through CCK (Complementary Code Keying).

The parameters chosen, in this work, to evaluate the quality of the communication in the wireless environment are:

- Received Signal Strength (RSS): it is a way to measure the power attenuation value of the signal. It is usually measured in the wireless interface chipset. In the IEEE802.11b standard, it varies from -255db to 0db (manufacturer dependent value).
- Link Quality: the correlation between the RSS and the (considered) ideal signal level.

#### A. Wireless channel characteristics

In this section, we show the causes of variations on the wireless channel and how they influence the forecast. Special attention is given to the variations that affect the received signal strength (RSS).

Forecasting in IEEE 802.11 WLANs is not an easy work, because the communication channel suffers interferences from the environment through refraction and reflection. Moreover, the 2.4GHz waves are absorbed by water, thus, absorbed by the human body. Because of environmental interferences and frequent variations on the channel quality parameters, it is difficult to identify a pattern that describes the behavior of the channel. In order to make a better evaluation of the channel characteristics, [7] divided the variations in two categories: temporal variations and space variations.

Temporal variations occur when the receptor (mobile device) is in some fixed position, so the variations are only time dependent. A description of the communication quality through time variation can be obtained from the RSS.

Space variations occur when the position of the receptor (mobile device) changes. Figure 1 shows the RSS behavior,

varying according to the space. The graph show results from real experiments.

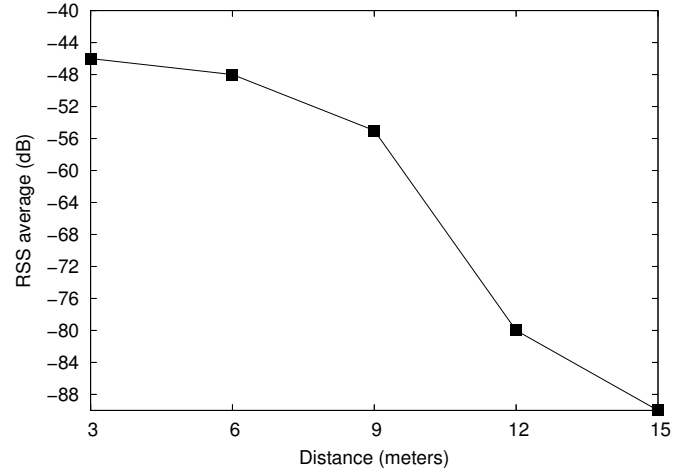


Fig. 1. Received signal strength indicator space variation

### IV. EXPONENTIAL SMOOTHING AND ITS VARIANTS

Exponential smoothing and its variants belong to a large class of forecast methods that deal with fluctuations on time series. The popularity of these methods is due to their precision and low computational effort. These methods forecast the next value on a time series by smoothing the curve passing through the observed data. Assuming that the extreme values of the series represent random fluctuations, these methods identify the basic pattern in the collected data and use it to forecast the future values. We shall present the Simple Exponential Smoothing (SES) [8], the Adaptive Exponential Smoothing (AES), also known as Trigg & Leach (TL) [5], and our proposed method, called the Natural Adaptive Exponential Smoothing (NAES).

#### A. Simple Exponential Smoothing (SES)

The Simple Exponential Smoothing (SES) method, like the Simple Moving Average method (SMA) [9], also smooths the abrupt behavior of the collected data, but, unlike the SMA method, it gives different weights to different points in the time series: points that are more recent in the time series have heavier weights [10]. The argument for this is based on the assumption that the more recent observations contain more information about the future and, therefore, are more relevant to the forecast. Box and Jenkins [8] set the following convex combination for the SES method:

$$\hat{Z}_{t+1} = \alpha Z_t + (1 - \alpha)\hat{Z}_t, \quad t = 1, \dots, N, \quad (1)$$

where  $Z_t$  is the observed data at time  $t$ ,  $\hat{Z}_{t+1}$  is the value computed by the SES method at time  $t + 1$ , and  $\alpha$  is the smoothing factor.

The SES method is, in its essence, a weighted mean where the heavier weights correspond to the more recent points in the time series, eliminating one of the disadvantages of the Simple

Moving Average method [9]. Equation 1 can be rewritten to  $\hat{Z}_{t+1} = \alpha e_t + \hat{Z}_t$ , where  $e_t = Z_t - \hat{Z}_t$  is the forecast error at step  $t$ . Thus, the new forecast is made by adding a multiple of the forecast error.

According to [9], the  $\alpha$  value determines the adjustment applied to the data. The lesser the value of  $\alpha$ , the bigger is the forecasting stability, since a lower value of  $\alpha$  implies in the attribution of a bigger weight to the less recent observations and, therefore, any fluctuation in the present contributes with lesser importance to the forecast. However, there is no good methodology to select an appropriate value to  $\alpha$ : it is usually found by trial and error [10]. This is the main disadvantage of this method.

The advantages of the SES method are, thus: simplicity; low computational effort; adaptability of the smoothing factor to each particular problem; and it is a one-step method, i.e., to compute  $\hat{Z}_{t+1}$  one only needs  $\hat{Z}_t$ .

### B. Adaptive Exponential Smoothing or Trigg & Leach(TL)

One of the weaknesses in the SES method is that the smoothing factor  $\alpha$  is considered constant throughout the whole series [5]. The TL method solves that weakness by adapting the value of  $\alpha$  accordly to variations on the basic pattern of the time series. When the system is unstable, the value of  $\alpha$  is set to be closer to one, meaning that the more recent points in the time series would be even more relevant to the forecast. Conversely, when the system is stable,  $\alpha$  is set to a lower value (closer to zero), so even the not so recent points in the time series are taken into account to the forecast.  $\alpha$  is defined, for each time step, by:

$$\alpha_t = \left| \frac{E_t}{M_t} \right|, \quad t = 2, \dots, N, \quad (2)$$

where  $E_t = \beta e_t + (1 - \beta)E_{t-1}$ ,  $M_t = |e_t| + (1 - \beta)M_{t-1}$ ,  $\beta$  is the variation speed factor of  $\alpha$  (normally 0.1 or 0.2) and  $e_t = Z_t - \hat{Z}_{t+1}$ , or either, the forecasting error in instant  $t$ .

This method has most of the advantages of SES with the additional one of dynamically adapting the  $\alpha$  through the series. It can be used with multiple smoothing factors [5], but the procedure to apply them is not very clear. The method showed in the next section resembles this method in the sense that it also has an adaptive smoothing factor  $\alpha$ .

## V. NATURAL ADAPTIVE EXPONENTIAL SMOOTHING (NAES)

Inspired by the basic ideas of the SES method and by the adaptive behavior of the TL method, our proposed algorithm, Natural Adaptive Exponential Smoothing, is presented in this section. It's name comes from the natural way it adapts the smoothing factor  $\alpha$ .

A way to measure the forecast error, is defining the distortion,  $\Delta$ , for a series of  $n$  values of  $Z$  as the sum of the squares of the differences between the observed values ( $Z$ ) and the computed ones ( $\hat{Z}$ ). That is:

$$\Delta = \sum_{i=1}^n (Z_i - \hat{Z}_i)^2$$

In contrast to the TL algorithm, which changes  $\alpha$  at each iteration, NAES changes  $\alpha$  in each interval of a chosen size  $S$  using an empiric constant  $C$  to quantify how much  $\alpha$  will change. For this, in each interval of size  $S$ , we generate three smoothing curves, using the values  $\alpha$ ,  $\alpha + C$ , and  $\alpha - C$ . We, then, calculate the distortions for each of these curves, and choose the smoothing factor  $\alpha$  corresponding to the smallest distortion. Algorithm 1 clarifies the steps.

**Input:** Time-series  $Z$

**Data:** Constant  $C$ , constant  $S$  and initial  $\alpha$

**Output:** Forecasting

**while**  $Z$  is not empty **do**

    Remove  $S$  elements from  $Z$ ;

    Make a simple exponential smoothing using  $\alpha$ ,  $\alpha + C$ ,  $\alpha - C$ ;

    From the forecasts generated from the three series

    calculate the distortions  $\Delta_\alpha$ ,  $\Delta_{\alpha+C}$ ,  $\Delta_{\alpha-C}$ ;

**if**  $\Delta_\alpha \leq \Delta_{\alpha+C}$  **e**  $\Delta_\alpha \leq \Delta_{\alpha-C}$  **then**

        |  $\alpha \leftarrow \alpha$ ;

**else**

**if**  $\Delta_{\alpha+C} \leq \Delta_{\alpha-C}$  **then**

            |  $\alpha \leftarrow \alpha + C$ ;

**else**

            |  $\alpha \leftarrow \alpha - C$ ;

**end**

**end**

**end**  
**Algorithm 1:** Natural Adaptive Exponential Smoothing

Like in the TL method, NAES sets the smoothing factor  $\alpha$  to a higher value when the system is unstable, and to a lesser one, when the system is stable. But since it varies  $\alpha$  only in every size  $S$  intervals, NAES tends to do this slowly, in a way that random fluctuations have less influence to the forecast.

### A. Choosing NAES parameters

In our algorithm, we have to define a good value for  $C$  and  $S$ . There was no special rule for finding good values, we applied trial and error. In our testbed, we've tried values 5, 10, 15, 30 and 60 for  $S$ , and 0.01, 0.05 and 0.10 for  $C$ . The best ones we're  $C = 0.05$  and  $S = 10$ .

## VI. RESULTS

In order to evaluate the NAES method, two types of tests were performed: tests performed with simulation, and tests performed, in real time, in a real environment.

### A. Real Environments

In order to evaluate the NAES method in real environments, we took into account real world situations such as obstacles, interferences from the environment, and the device's mobility. The tests were performed in three kinds of environment: In the first one, there was a line-of-sight (LOS) between the access point (AP) and the mobile device. In the second environment, there was an obstacle between the mobile device and the AP (a non-line-of-sight situation - NLOS). In the third one, the

mobile device moved randomly in a building (a 400m<sup>2</sup> floor in the Statistics and Computer Sciences Department building, in Itaperi campus of the State University of Ceará).

In each testbed, the measurements were performed every second, during a time range of 3600 seconds (1 hour). The measurements were collected through a HP NX 9010 notebook running wireless tools [11] on a 2.4.32 Linux kernel. The algorithm was implemented in application-level, using C language. Then the TL and the NAES methods were applied to the collected data. The parameters chosen for the NAES algorithm were, 0.05 for the  $C$  constant, 10 steps for the slice  $S$  and 0.5 for the initial  $\alpha$ .

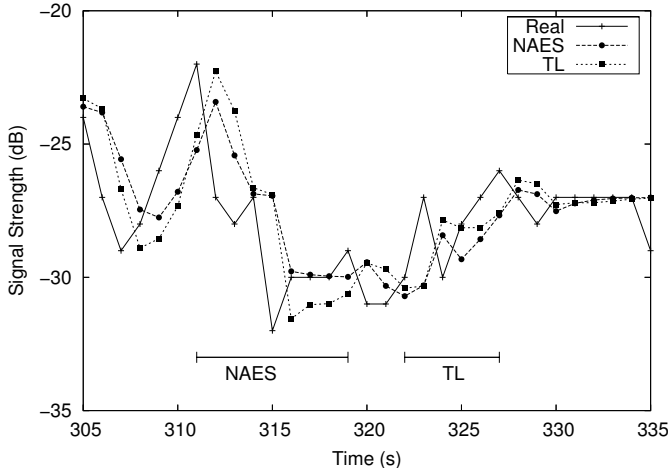


Fig. 2. Comparative analysis of the RSS real values and foreseen values

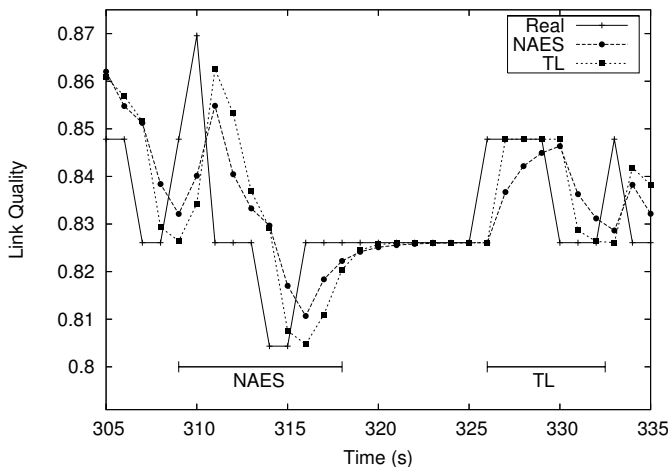


Fig. 3. Comparative analysis of the link quality real values and foreseen values

Figures 2 and 3 shows the comparison between the measurements of the RSS and the values computed by the TL and the NAES methods, in the mobility scenario. For the sake of clarity, only the period from 305 to 335 was selected for display. Note that both TL and NAES methods give results close to the real values. We highlighted in Figures 2 and 3

the intervals where NAES’s predictions were better than the ones made by the TL method, and the converse. The NAES method tends to perform better when the data changes are not too abrupt, the converse holds when the changes are abrupt. This is due to the fact that, in the TL method, the smoothing factor  $\alpha$  can reach values close to one faster than in the NAES algorithm, following the abrupt data fluctuations better.

In order to compare TL and NAES methods better, a least square error analysis was made for a time period of one hour and for the three environmental scenarios. The results are shown in Table I and Table II.

TABLE I  
LEAST SQUARE ERROR - RSS

Method	Mobility	LOS	NLOS
NAES	3.205048770	3.630953116	3.329427908
TL	1.939613290	32.063906250	3.988009000

TABLE II  
LEAST SQUARE ERROR - LINK QUALITY

Method	Mobility	LOS	NLOS
NAES	0.000000753	0.000984328	0.000050041
TL	0.000070510	0.002898746	0.000141139

In Table I, the errors for the NAES method were smaller in LOS and NLOS scenarios, but larger for the Mobility scenario. This is due, again, to TL’s ability to change  $\alpha$  faster, following abrupt changes – a mobility scenario characteristic – better.

As for the link quality, the NAES method gave better results in all three scenarios, as shown in Table II. This was unexpected, since the link quality should also presents abrupt changes in the Mobility scenario. To validate the foreseen values of link quality, the link quality definition was taken in account. This definition correlates the incoming signal to the ideal signal. It’s important to know that the signal DSSS measure differs to the Signal Strength measure. Thus, the proposed algorithm showed a minor error in all the situations, showing itself to be efficient forecasting the link quality.

B. Simulation

TABLE III  
LEAST SQUARE ERROR - RANDOM GAUSSIAN DISTRIBUTION

Method	First	Second	Third
NAES	125594.80	131827.67	464060739.44
TL	135693.10	134270.25	466221798.81

To compare the methods in the simulated environment we generated three random gaussian distributions. Gaussian distribution was chosen due to the gaussian behavior of channel quality parameters [12]. The distributions were generated using Box-Muller transformation [13]. In the first one, we used

mean 0 and deviation 1. In the second one, the mean is 0.57, and the deviation is 0.20 (mimicking link quality on mobility scenario). In the third one, the mean is -66, and the deviation is 29 (mimicking RSS on mobility scenario). As seen in table III NAES was slightly better for all tests.

## VII. CONCLUSION AND FUTURE WORK

This work proposes a forecast method, named NAES, for IEEE 802.11b WLANs channel behavior time series. The NAES method is based on the SES and TL exponential smoothing methods. It proved to be more efficient when compared to the TL adaptive method, with the additional advantages of simplicity and low computational effort. These characteristics make the method suitable for implementation in a great variety of mobile devices that belong to Radio Frequency technologies. However, in simulated environments, it is not efficient as TL, because of its natural way of adapting  $\alpha$ .

In a future work, we intend to evaluate new mobile test scenarios, including one with multi-varied (video, audio and data) traffic being generated in real-time [2]. We plan to validate the NAES algorithm, comparing it to others. We also plan to propose and implement a new forecast method based on time series which adopts linear models such as AR, ARMA, and ARIMA [14] and test it within a WLAN environment.

## REFERÊNCIAS

- [1] N. S. Yoshikazu Akinaga, Shigeru Kaneda and A. Miura, "A Proposal for a Mobile Communication Traffic Forecasting Method Using Time-series Analysis for Multi-variate Data," in *Globecom*, 2005.
- [2] C. Na, J. Chen, and T. Rappaport, "Hotspot Traffic Statistics and Throughput Model for Several Applications," in *Global Telecommunications Conference, 2004. GLOBECOM'04. IEEE*, 2004.
- [3] M. A. C. Jorge L. C. e Silva and P. R. F. Cunha, "ProgCon: um Sistema para Prognóstico de Congestionamento de Tráfego de Redes," in *SBRC2005*.
- [4] K. L. Dias, S. F. L. Fernandes, and D. F. H. Sadok, "Controle de Admissão e Reserva de Recursos em Redes Móveis Utilizando Previsão de Séries Temporais," in *SBRC2004*, 2004.
- [5] D. W. Trigg and A. G. Leach, "Exponential Smoothing with an Adaptive Response Rate," no. 1, pp. 53–59, mar 1967.
- [6] R. P. C. Ricardo A. Rabelo Oliveira and A. A. F. Loureiro, "Monitoramento de Eventos em Redes Sem Fio Utilizando EWMA e Wavelets," in *SBRC 2005*, maio.
- [7] M. A. A. Y. A. Rehim, "Horus: A WLAN-Based Indoor Location Determination System," Ph.D. dissertation, University of Maryland, College Park, 2004.
- [8] G. Box and G. Jenkins, *Time Series Analysis, Forecasting and Control*. Holden-Day, Incorporated, 1990.
- [9] P. A. Morettin and C. M. C. Toloí, *Análise de Séries Temporais*. BLÜCHER, 2004.
- [10] S. C. Wheelwright and S. Makridakis, *Forecasting Methods for Management*. N.Y, Wiley, 1985.
- [11] J. Tourrilhes, "Wireless tools for linux," Available on: [http://www.hpl.hp.com/personal/Jean\\_Tourrilhes/Linux/Tools.html](http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html), 2007, release March, 27th 2007.
- [12] T. Rappaport, *Wireless communications*. Prentice Hall PTR Upper Saddle River, NJ, 2002.
- [13] G. Box and M. Muller, "A Note on the Generation of Random Normal Deviates," *The Annals of Mathematical Statistics*, vol. 29, no. 2, pp. 610–611, 1958.
- [14] E. Bradley, "Time-series Analysis," *Intelligent Data Analysis: An Introduction*. Springer, 2000.