Wireless Network Delay Estimation for Time-Sensitive Applications

Rafael Camilo Lozoya Gámez, Pau Martí, Manel Velasco and Josep M. Fuertes
Automatic Control Department
Technical University of Catalonia
e-mail: {$camilo.lozoya, pau.marti, manel.velasco, josep.m.fuertes$}$@upc.edu$

Research report ESAII RR-06-12

July 27, 2006

Abstract

Wireless Local Area Networks (WLAN) based on 802.11b technology (commercially known as WiFi) have become, in the last few years, quite popular and widespread. The nature of the radio channel and the access to the shared resource cause variable available bandwidth, and variable packet delay and loss rate. This may prevent the correct operation of the networked applications, such as multimedia or control applications.

For example, in-home entertainment devices are increasingly using WiFi for streaming video and audio. Due to the unstable characteristics of the radio link, Quality-of-Service (QoS) problems quickly arise. In particular, it is desirable to have mechanisms for estimating the available bandwidth in order to be able to adapt the streaming content, thus providing quality guaranties.

In the automation area, there is a clear trend promoting the use of wired and wireless ethernet in the factory floor. Moreover, closing control loops over wireless networks is rising interest. However it is known that the delay introduced by the network may degrade control performance. Therefore, a good estimation of the network delay will facilitate robust controller designs.

The Round Trip Time (RTT) delay measurement can be used to infer the state of a network connection between two or more hosts. RTT delay can be used in a WiFi network, as a parameter to measure the network load and also to evaluate the available bandwidth. This work presents a model that implements six different techniques focused on the estimation of the RTT delay in a network. The RTT delay values are generated based on characteristics already observed in WiFi networks. The model includes three algorithms based on statistical formulas (Mean Value Estimation Algorithm, Median Value Estimation Algorithm, Max Value Estimation Algorithm), one algorith based on the Markov Chain algorithm, one algorithm based on the Exponential Averaging technique, and one algorithm proposed in this work based on the Kalman Filter algorithm. The Integrated Absolute Error (IAE) and the Integrated Square Error (ISE) calculations are used in this model, in order to evaluate the performance of each algorithm.

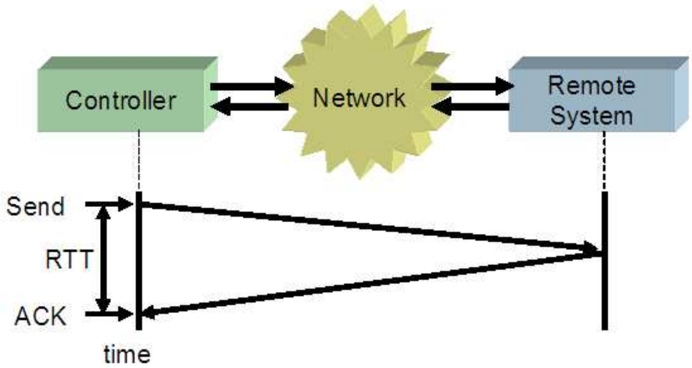

Figure 1: The Round Trip Time (RTT) represents a way to measure the network load.

1 Introduction

Wireless Local Area Networks (WLAN) based on 802.11b technology (commercially known as WiFi) have become, in the last few years, quite popular and widespread [2]. From a QoS and traffic management point of view, the main problem with WiFi networks is the relative low capacity of the shared radio channel. The nature of the radio channel and the access to the shared resource cause variable packet delay and loss rate.

In WiFi networks, Round Trip Time (RTT) delay measurements constitute the basic feedback information that the end hosts can use to infer the state of the network connection between them [3]. RTT is defined as the time in between, when the data packet is sent and when the acknowledgment is received. The RTT delay measurement is used to infer the rate at which a real time connection between two hosts can be initiated, so that it does not disrupt existing connections.

This work presents an analysis of different techniques focused on the estimation of RTT delay for data transmission. The estimation techniques presented in this work, includes four algorithms proposed by Chow [6] (Mean Value Estimation Algorithm, Median Value Estimation Algorithm, Max Value Estimation Algorithm and Markov Chain Estimation Algorithm) one algorithm proposed by Lennvall [5] (Exponential Averaging Algorithm), and one model proposed in this work based on the Kalman Filter algorithm. Kalman Filter algorithm has already been used in the prediction of a 3-D motion of a moving object in order to forecast its video traffic in a videoconference system [4]. The six algorithms are implemented in a Matlab model [7], so the analysis of the algorithms is based on the discussion of the simulation results provided by the Matlab model.

The six algorithms that conforms the model, are simulated in an environment which presents random variations in the available bandwidth. The traffic load is defined by the network delay. Network delay is measured by the Round Trip Time (RTT), recorded from the time when the server (controller) sends a packet out to the time the controller receives the acknowledgement. RTT depends on network equipment, throughput and the congestion condition on the network during packet transmission (see Figure 1).

The RTT delay distribution, in a Wireless Local Area Network (WLAN), is usually unimodal and asymmetric and has a long tail on the right hand side [3]. In Gunawardena *et al* [1], the behavior of the RTT delay, during measurements obtained in a Voice over IP (VoIP) connection, presents a density distribution with an unimodal trend. In order to model these characteristics we have chosen a Gamma Distribution to generate the random RTT delay values to evaluate the proposed models.

2 Algorithms Description

This section describes the characteristics of the algorithms to be evaluated. This information serves as a reference for the implementation of the five different algorithms that comprise the RTT delay model.

The first three algorithms (Mean Value Estimation, Median Value Estimation and Max Value Estimation) are based on simple statistical formulas, and require to store historical data for the calculations. Meanwhile the Markov Chain, the Exponential Averaging and the Kalman Filter algorithms, do not require to store historical data, since the calculations are based only on the system current state. The main difference between this last three algorithms is that, the Kalman Filter implements a dynamic closed loop control system, meanwhile the Markov Chain takes previously defined static information as the basis of the algorithm, finally the Exponential Averaging algorithm generates the estimated value based on a combination of the previous estimation plus the last bandwidth measurement.

2.1 Mean Value Estimation Algorithm

This algorithm predicts the current RTT delay by the mean value of last w measured RTT delays. Where w is known as the window size, and its value defines the amount of previous data that needs to be kept. If w is high, more resources are required to keep the data, and as a result more resources are needed to implement the algorithm.

$$\hat{\tau}_{k+1|w} = mean\{\tau_k, \tau_{k-1}, ..., \tau_{k-w+1}\} \tag{1}$$

2.2 Median Value Estimation Algorithm

This model predicts the current RTT delay by the median value of last w measured RTT delays.

$$\hat{\tau}_{k+1|w} = median\{\tau_k, \tau_{k-1}, ..., \tau_{k-w+1}\} \tag{2}$$

2.3 Max Value Estimation Algorithm

This algorithm chooses the larger value between the mean and median of last w measured RTT delays, and uses it as the next estimated network delay value.

$$\hat{\tau}_{k+1|w} = max\{mean(\tau_k, \tau_{k-1}, ..., \tau_{k-w+1}), median(\tau_k, \tau_{k-1}, ..., \tau_{k-w+1})\} \tag{3}$$

2.4 Markov Chain Estimation Algorithm

Markov Chain algorithm predicts the delay range by classifying the Markov states and its state-transmission matrix. The Markov states at time k are denoted as $\{1,...S\}$, where S is the total number of states. The state-transmission matrix, denoted as R_{SxS}, captures the dynamic changes from state to state, where its values are calculated from the *a priori* probability distribution of previous dynamic state changes. For more information regarding this algorithm see paper [8].

2.5 Exponential Averaging Estimation Algorithm

The Exponential Averaging algorithm is a technique used to examine and averages a sequence of values along a time series which enable to make a prediction based on previous prediction as well as the current network load. The bandwidth prediction is defined by the following formula.

$$P_k = \alpha BWT_k + (1-\alpha)P_{k-1} \tag{4}$$

Where P_k is the future, P_{k-1} is the previous prediction, BWT_k is the current bandwidth measurement, and α is a constant used to determine how important the history versus the current measurement is in the prediction. See paper [5] for more information about this algorithm.

2.6 Kalman Filter Estimation Algorithm

The Kalman Filter is essentially a set of mathematical equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance. For more details about the algorithm and the mathematical fundaments of the Kalman Filter refers to [9].

The Kalman Filter is used to estimate the state x, of a discrete process governed by the linear stochastic difference equation.

$$x_k = Ax_{k-1} + Bu_k + w_k \tag{5}$$

Where the x value is measured through the z value defined by the following equation.

$$z_k = Hx_k + v_k \tag{6}$$

The process and measurement noise are defined by the w_k and v_k values respectively. It is assumed that they are independent, with a zero mean and with a covariance value of Q and R respectively.

$$p(w)\tilde{}M(0,Q) \tag{7}$$

$$p(v)\tilde{}N(0,R) \tag{8}$$

The Kalman Filter estimates a process by using a form of feedback control. The filter estimates the process state at some time and then obtains feedback in the form of noisy measurements. The equations for the Kalman Filter are divided in two groups: time update equations (predictor) and measurement update equations (corrector).

The time update equations are responsible of the prediction of the current state and error covariance, in order to obtain the *a priori* estimates. These equations conform the predictor stage of the filter.

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \tag{9}$$

$$P_k^- = A^2 P_{k-1} + Q \tag{10}$$

Where $\hat{x}_k^-$ represents the *a priori* estimate of the process current state, and P_k^- is the *a priori* estimate of the covariance error.

The measurement update equations are responsible for the feedback information and for the incorporation of the new measurement into the *a priori* estimate to obtain an improved *a posteriori* estimate. These equations conform the corrector stage of the filter.

$$K_k = \frac{HP_k^-}{H^2 P_k^- + R} \tag{11}$$

$$P_k = P_k^-(1 - HK_k) \tag{11}$$

$$\hat{x}_k = x_k^- + K_k(z_k + H\hat{x}_k^-) \tag{13}$$

Where K_k is the Kalman constant, P_k is the *a posteriori* estimation of the covariance error, and $\hat{x}_k$ represents the *a posteriori* estimate of the process current state.

3 Network Delay Model

This section describes the characteristics of the Matlab model that was developed based on the six algorithms described in the previous section. The purpose of the model is to obtain performance information of the different algorithms, through the simulation process.

The overall model is divided in six different stages (see Figure 2). The sequence of tasks executed during each stage is described below.

3.1 Set Up Initial Parameters

During the first stage the initial parameters of the model are defined. The main parameters are: the number of estimation events for each execution cycle, the number of cycles that comprise the simula-

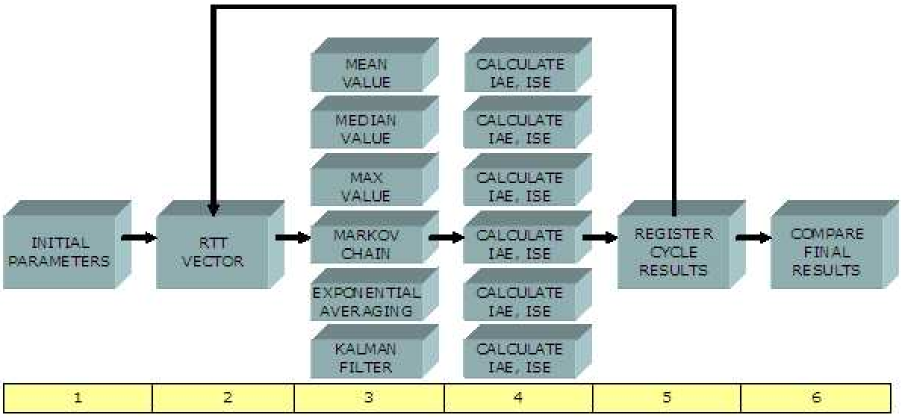

Figure 2: The complete Matlab model is composed by six stages.

tion. Also in this stage the window size, used for the statistical algorithms, is defined. It is important to highlight that this parameter defines the amount of memory resources required by the model. In the case of the Exponential Averaging algorithm the α parameter is defined.

3.2 Generate RTT Delay Vector

In the second stage, a vector, that contains the RTT delay values, is created. The size of the vector is equal to the number of estimation events. The RTT delay values are generated using a Gamma distribution based random number generator function (see Figure 3). Gamma distribution is used since the behavior of the WiFi network delays, has similar characteristics to this type of density distribution.

3.3 Algorithm Execution

In third stage, the six algorithms are executed in order to estimate the RTT delay. Every algorithm uses the RTT delay vector created in the second stage as the current values of the model. In the Markov Chain estimation model, a 5x5 transition matrix is created, with values that correspond to a Gamma distribution model. In the Kalman Filter estimation model the peak value of the Gamma distribution is used as the initial guess for the model current state. In the Mean Value, Median Value, Max Value and Exponential Averaging algorithms also the peak value of the Gamma distribution is used as the initial parameter.

3.4 Performance Measurement

During the fourth stage, the performance of each algorithm is measured. The integral absolute error (IAE) and the integral square error (ISE) are used in order to measure how close the estimated value is with respect the current value. Both IAE and the ISE are used as performance index in the proposed model.

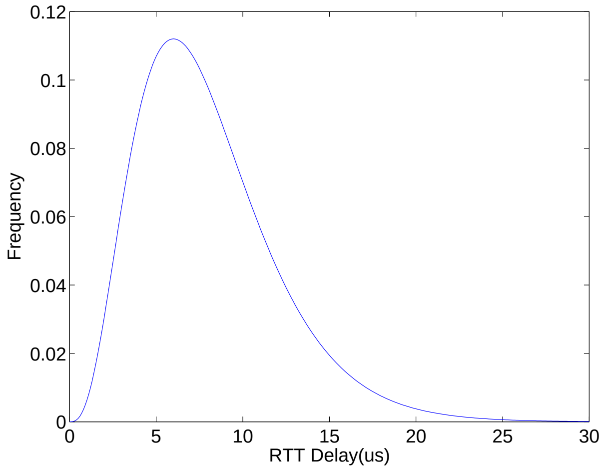

Figure 3: RTT delay random values are generated based on a Gamma distribution. The RTT range is from 0 to 30 μs.

3.5 Register Cycle Results

The IAE and the ISE values for each algorithm are stored during this stage. The reason is to execute this cycle several times in order to increment the amount of information during the simulation. The number of cycles to be executed during the simulation is defined in the stage 1.

3.6 Compare Final Results

Finally the results of the simulation are compared, through a series of graphs, that help to understand the performance of each algorithm during different scenarios.

4 Simulation Results

The discussion of the data obtained, as a result of the simulations conducted on the proposed model, is divided in three parts. In the first part we analyze how the key parameters of each algorithm impact on the performance of the model. In the second part we analyze how the different algorithms behaves during the estimation process. In the last part, we focused in the discussion of the performance of the six algorithms.

4.1 Parameters Tuning

4.1.1 Mean Value, Median Value and Max Value Algorithms

In the statistical based algorithms, the window size parameter directly defines the performance of the algorithms. In order to improve the performance of the Mean, Median and Max Value algorithms, the window size needs to be increased. This situation, in practical terms, requires increment the amount of memory resources of the system. Figure 4, shows how the performance of the statistical algorithms (Mean, Median and Max Value) improves when the window size increases and the number of events per cycle remains fixed. The ISE performance index is used to measure the performance. So in order to obtain the best performance in these algorithms the window size must have the highest possible value.

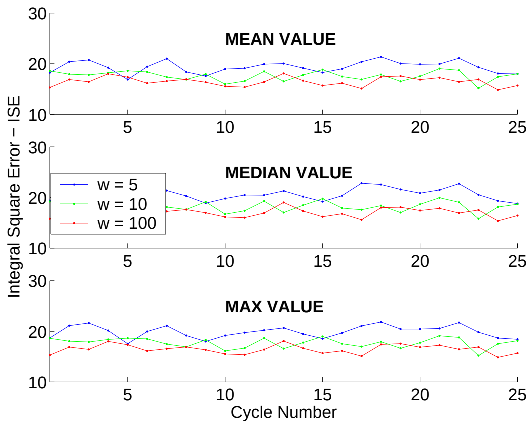

Figure 4: Behavior of the statistical algorithms with different window size value.

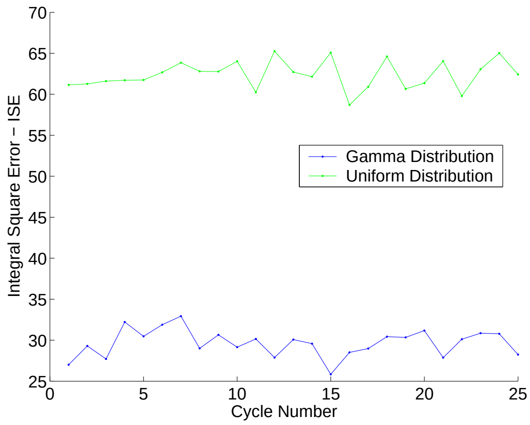

Figure 5: Performance comparison of the Markov Chain algorithm when the transition matrix is based on Gamma distribution versus a transition matrix based on Uniform distribution.

4.1.2 Markov Chain Algorithm

In the Markov Chain algorithm, the key parameter is the transition matrix. This matrix establishes a static probability of the estimated next state, based on the current one. One approach, in order to tune this parameter is to obtain the probability values from the characteristics of the density distribution function, in this case we are using a Gamma distribution. Figure 5, presents the results of the simulation of the Markov Chain algorithm using two different transition matrix, one based on the Gamma distribution density function and the other based on a Uniform distribution. It can be seen that the performance of the transition matrix based on Gamma distribution superates the Uniform distribution. The ISE performance index is used to measure the performance.

4.1.3 Exponential Averaging Algorithm

The α value in the Exponential Averaging algorithm defines which criteria has more influence in the estimation process: the last bandwidth measurement or the previous prediction value. A simulation process was conducted, for the Exponential Averaging algorithm in which different α values were used. As it can be seen in figure 5, the α value of 0.1 provides the best performance on the algorithm using the ISE performance index. This means that the last bandwidth measurement value must have more weight in the formula, in order to improve the performance. But it is important to notice that the previous prediction value must have a minimal and non-zero influence in the formula.

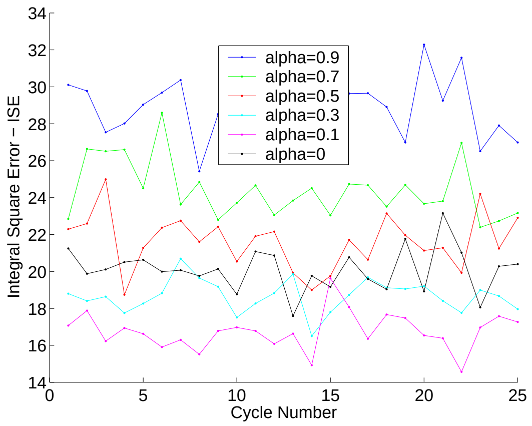

Figure 6: Simulation results for the Exponential Averaging algorithm with different α values.

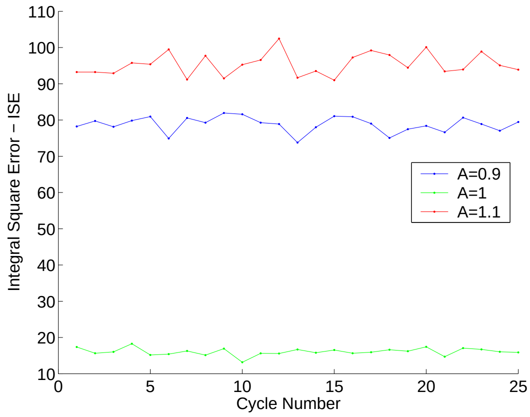

Figure 7: Simulation results for the Kalman Filter algorithm with different constant feedback A values.

4.1.4 Kalman Filter Algorithm

The Kalman Filter algorithm is composed by different parameters. But the key parameters in order to tune the algorithms are following. The constant feedback parameter A (from equation 5) must have a value of 1, if $A <> 1$, the performance is very low (see figure 6). The initial value of P_{k-1} (from equation 10), also known as P_0, must have a value different from 0, If $P_0 = 0$, would cause the filter to initially and always believe $\hat{x}_k^- = 0$, and the filter would never converge. Finally, in order to minimize effect of the process variance noise the Q value from equation 7 is small but non-zero. It is important to mention that the Kalman Filter algorithm is the only one that takes in consideration the effect of the noise.

4.2 Estimation Process

Figure 7 shows the estimated values provided by the different algorithms, in response to a random sequence of 30 events. The same random sequence that represents the current RTT delay is used in each one of the six algorithms.

It can be observed that the current RTT delay values are concentrated between 4 and 10 μs [1]. This corresponds to the fact that we are using the Gamma distribution, for random number generator, presented in figure 3.

Mean, Median and Max Value algorithms practically produce the same estimated values, always between the range of 4 to 10 μs. So we can assume that the statistically based algorithms have the same behavior in the estimation process, when a Gamma distribution is used for the random number generator, that represents the RTT delay of a WiFi network.

The Markov Chain algorithm produces some predicted values outside the range of 4 to 10 μs. So this algorithm is the only one that takes in consideration the low density values in the prediction

[1] This range was defined just for simulation purposes only, it does not correspond necessarily to specific values measured in a network

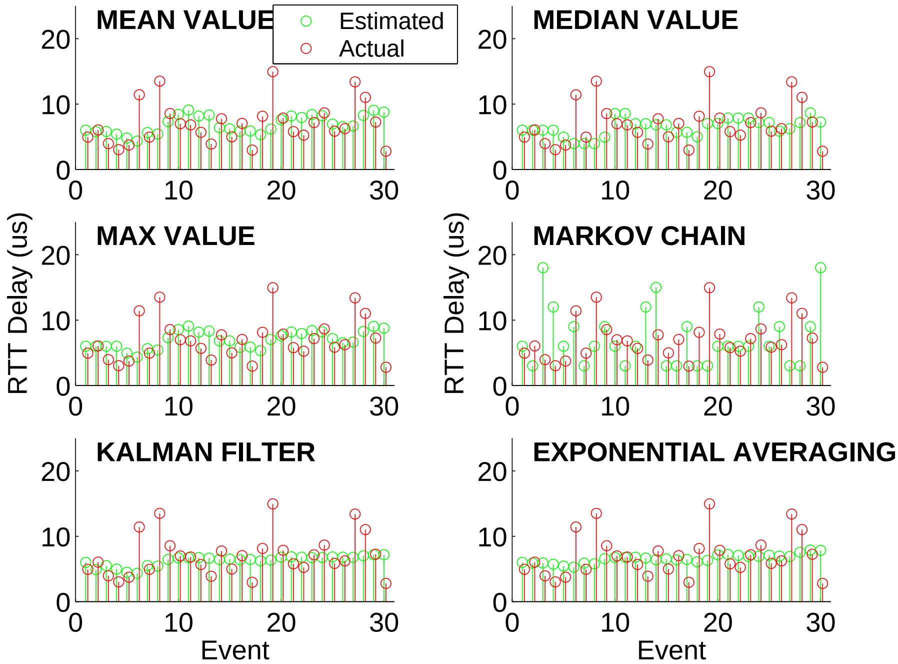

Figure 8: Simulation results to estimate the RTT delay using six different algorithms.

process.

The Kalman Filter algorithm presents estimated values also within the range of 4 to 10 μs. The difference of this algorithm, in comparison with the statistical based algorithms, is the fact that variation between the current estimated value and the previous one is smaller. This situation, makes that this algorithm produces the smoothest response.

The Exponential Averaging algorithms also presents a response very similar to the Kalman Filter, even though its mathematical background in the algorithms is different.

4.3 Algorithms Performance

The Integrated Absolute Error (IAE) and the Integrated Square Error (ISE) calculations are used in this model, in order to evaluate the performance of each algorithm. So the performance index of this model is integrated by the IAE and ISE.

During the execution of different simulation scenarios, it was identified that the duration of the cycle has an impact over the performance of the algorithms. So four simulation scenarios were evaluated for the proposed model. We consider four kind of simulation cycles: a short cycle, a medium cycle, a large cycle, and an extra-large cycle. The short cycle is conformed by the execution of 10 prediction events, the medium cycle equals 100 prediction events, the large one is equivalent to 1000 events, and the extra-large contains 10000 prediction events. In order to evaluate the results of the different scenarios, 25 cycles of each kind were conducted during the simulation process.

If the number of events per cycle is very low (equals to 10), the six algorithms have a very similar

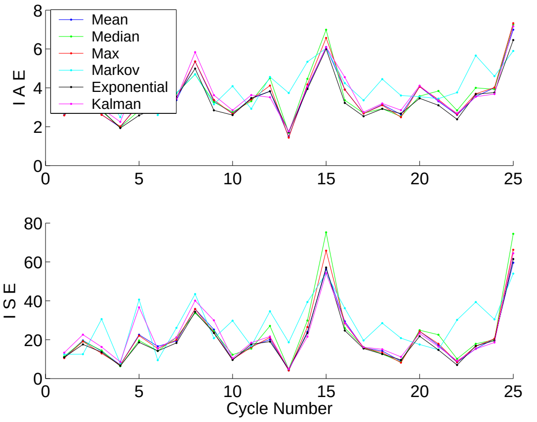

Figure 9: Algorithms performance in a simulation of 25 cycles with 10 events per cycle.

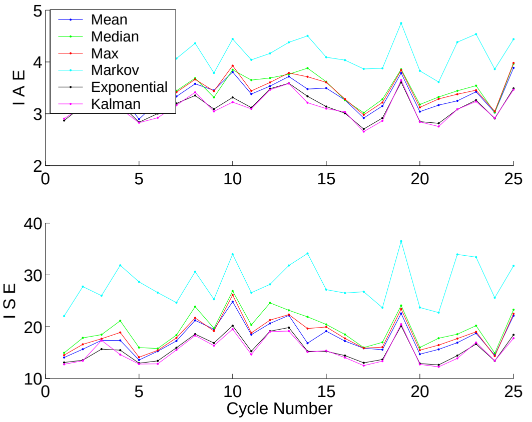

Figure 10: Algorithms performance in a simulation of 25 cycles with 100 events per cycle.

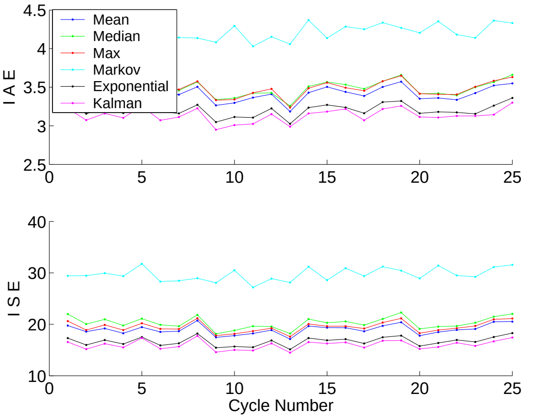

Figure 11: Algorithms performance in a simulation of 25 cycles with 1000 events per cycle.

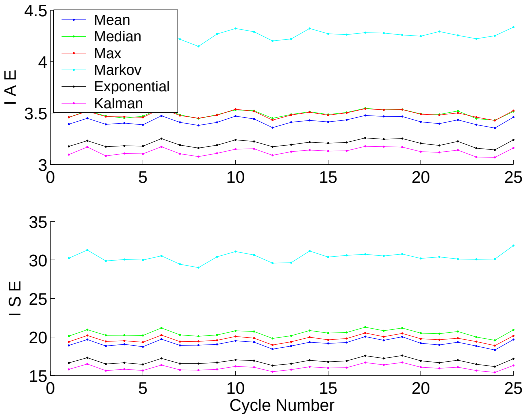

Figure 12: Algorithms performance in a simulation of 25 cycles with 10000 events per cycle.

performance, including the Markov Chain algorithm, see figure 8. Although it is desirable to manage more estimation events to evaluate the network transactions, it is interesting to observe the similarity in the performance of the algorithms, under this circumstances.

Figure 9 shows the results of a simulation of 25 cycles with 100 events for each cycle. We notice that, with the increment of the number of estimation events, the algorithms can be classified in three categories based on the performance. Kalman Filter and Exponential Averaging provides the best performance, Markov Chain has the worst, and the statistical based algorithms are in the middle, close to the Kalman Filter and Exponential Averaging.

If we extend the number of events from medium to large, the difference in the IAE and the ISE between Kalman Filter and Exponential Averaging algorithms with respect the Mean, Median and Max Value algorithms is increased. This situation is showed on figure 10. The performance of Mean, Median and Max Value is degraded, because the relative value of the window size with respect the number of events decreases, also the Kalman Filter and Exponential Averaging performances presents improvements.

Although, the performance of the Kalman Filter is similar with respect the Exponential Averaging algorithm, still the Kalman Filter presents the best performance, and this situation is more visible when the number of events is extra-large see figure 11).

Based on these results, we can conclude that the Kalman Filter and Exponential Averaging algorithms performances are not affected negatively by the number of events, this represents a key advantage of these algorithms, since it is highly probable that the WiFi network can present a behavior where the amount of transactions in a short period of time can be high. In other words, the scenario of larger number of events is more suitable to represent the traffic load of a WiFi network.

Comparing the Kalman Filter performance versus the Exponential Averaging, we notice that the first one presents a better performance. But the advantage of the second one is that the algorithm is simpler.

5 Conclusion

In this document a Matlab model is proposed, in order to estimate the RTT (Round Trip Time) delay in a data transmission WiFi network. The model compares the performance of three statistically based algorithms (Mean Value, Median Value, Max Value), one Markov Chain algorithm and finally a Kalman Filter algorithm. The delay RTT values are randomly generated based on a Gamma distribution, since this behavior is presented in different measurement on the WiFi networks. As a result the best performance is obtained from the Kalman Filter algorithm, and very close in second place the Exponential Averaging. The Exponential Averaging algorithm presents the advantage that is simpler than the Kalman Filter, even though its performance is slightly below than the Kalman. The advantage of the Kalman Filter and Exponential Averaging over the other algorithms increases when the number of estimation events in a cycle is larger. The statistical algorithms has a similar performance among them, and the performance has a strong dependency of the window size value used in order to perform their calculations.

References

[1] Corvaja, R., Zanella, A., Dossi, M., Tontoli, A., Zennaro, P., (2004) "Experimental Performance of the Handover Procedure in a Wifi Network", *Proceedings of WPMC04*, Abano Terme (Padova), Italy.

[2] Franceschinis, M., Mellia, M., Meo, M., Munafo, M., (2005) "Measuring TCP over WiFi: A Real Case", *WinMee: 1st Workshop on Wireless Network Measurements*, Riva del Garda, Italy, April 3, 2005.

[3] Gunawardena, D., Key, P., Massoulie, L., (2003) "Network Characteristics: Modelling, Measurements and Admision Control", *IWQoS 2003*, Monterey, June 2003.

[4] Jiong, S., Haibo, L., (2004) "3-D Physycal Motion-Based Bandwidth Prediction for Video Conferencing", *IEEE Transactions on Circuits and Systems for Video Technology, Vol 14., No 5*, May 2004.

[5] Lennvall, T., Fohler, G., (2005) "Providing Adaptive QoS in Wireless Networks by Traffic Shaping", *Resource management for media processing in networked embedded systems (RM4NES)*, Eindhoven, Netherlands, March, 2005.

[6] Li, Z., Vanijjirattikhan, R., Chow, M.Y., Viniotis, Y., (2005) "Comparison of Real-time Network Traffic Estimator Models in Gain Scheduler Middleware by Unmanned Ground Vehicle Network-Based Controller", *Proceedings of the 31st Annual Conference of the IEEE Industrial Electronics Society (IECON'05)*, Raleigh, North Carolina, Nov. 6-10, 2005.

[7] Matlab & Simulink Products, (2006) http://www.mathworks.com, The MathWorks Inc, 1994-2006.

[8] Nilsson, J., Bernhardsson, B., (1997) "LQG Control Over Markov Communication Network", *Lund Institute of Technology*, http://www.control.lth.se.

[9] Welch, G., Bishop, G., (2001) "An Introduction to the Kalman Filter", *University of North Carolina at Chapel Hill*, http://www.cs.unc.edu.

[10] Wust, C., Steffens, L., Verhaegh, W., Bril, R., Hentschel, C., (2005) "QoS Control Strategies for High-Quality Video Processing", *Real-Time Systems, Springer Science + Business Media*, Netherlands 2005.