Proceedings of 2011 8th Asian Control Conference (ASCC)
Kaohsiung, Taiwan, May 15-18, 2011

Delay Modeling and Estimation of a Wireless Based Network Control System

Jianjun Bai
Institute of Cyber-System and Control, Yuquan Campus
Zhejiang University
Hangzhou, China 310027
Email: baijianjun83@gmail.com

Minyue Fu
School of Electrical Engineering and Computer Science
University of Newcastle
Callaghan, NSW 2308 Australia
Email: minyue.fu@newcastle.edu.au

Hongye Su
Institute of Cyber-System and Control, Yuquan Campus
Zhejiang University
Hangzhou, China 310027
Email: suhy@supcon.com

***Abstract*—In this paper, a wireless based network control system is considered. Poisson process models are proposed to approximate the transmission delays for both single-hop and multi-hop wireless transmissions. State estimation is done in the real wireless network and a simulated network using the Poisson process model. The estimate errors in the two cases match well, which verifies the validity of the delay model.**

I. Introduction

Wireless communication networks have been widely used for online monitoring and control. As an extension to or as an alternative for the traditional wired networks, there are many attractive advantages such as mobility, flexibility, scalability, easy installation, and reduced long-term cost in mobile environments [1, 2]. However, it is more challenging since wireless channels are prone to possible transmission errors caused by either channel outages or interferences.

When networks are introduced into the control systems, new problems appear, including packet losses and network-induced delays [3]. As is well known, time delay may reduce the performance of the system or even destabilize the system [4]. We have to consider network induced delays when designing control systems. To this end, it is necessary to properly model network delays.

In the literature, many results are available about time delay modeling. Kleinrock [5] first evaluated the CSMA mode and the average delay was studied in order to get a tradeoff between the average delay and the throughput of the network. In [6], a cluster-tree topology wireless network was studied and packet service time for the uplink and downlink were considered to prevent the networks to become saturated. Multi-hop delays were studied in [7] to evaluate network performances. A trace-based approach was used in [8] to reproduce the characteristic of a real network by simulation. However, in all the results above, delays are considered from the viewpoint of "networks" and it is hard to use these delay models in designing a network control system. In this paper, we will consider delays from the viewpoint of "control" and a Poisson process is used to model the network-induced delays. Both single-hop and multi-hop delays are considered. From experiments, we find that this delay model fits the experiment results well. Finally, Kalman filters are used for both the real wireless network system and a simulated network control system based on the Poisson delay model. The estimated errors in both cases match well, which further validates the delay model.

II. Experimental Setup

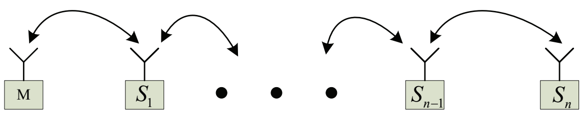


Fig. 1. Topology of a multi-hop network

Experiments are carried out using the multi-hop testbed as shown in Fig.1, in which M stands for the master node and S_i, $i = 1, ..., n$ stand for the slave nodes. When there is only one slave computer, the multi-hop network reduces to a single-hop network as shown in Fig.2. Each wireless node is composed of

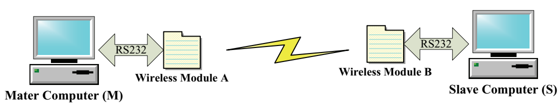


Fig. 2. Single-hop network

a computer and a wireless module compliant to the IEEE 802.15.4 standard. The computer communicates with the wireless module via a serial port.

The Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism is used to access the communication channel in the wireless networks. In the experiment, unslotted CSMA/CA is used [9]. Two parameters need to be predefined: NB and BE. NB is the number of times the CSMA/CA algorithm is required to backoff while attempting the current transmission. It is initialized to zero before every new transmission. BE is the backoff exponent, which is related to how many backoff periods a device shall wait before attempting to access the channel. The MAC layer will delay for a random number of backoff periods in the range of 0 to $2^{BE} - 1$ before requesting the physical layer to perform a CCA (Clear Channel Assessment). If the channel is assessed to be busy, the MAC layer will increase both NB and BE by one, provided that BE is no more than the $macMaxBE$. If BE is less than or equal to $macMaxCSMABackoffs$, another CCA is performed after a random waiting time; else the CSMA/CA will terminate with a Channel Access Failure status. If the channel is idle, the MAC layer starts transmission immediately.

III. Delay Models

A. Single-hop Delay

Consider the single-hop network shown in Fig.2. In the experiment, the round trip time delay is recorded. The master computer (M) sends a data request command to the slave computer (S) once every second and records the sending time T_1. The slave computer sends a reply packet back as soon as the command is received. When the master computer receives the reply packet, the receiving time T_2 is recorded and the round trip time delay T_{RD} is calculated as $T_{RD} = T_2 - T_1$.The packet sending and receiving process is shown in Fig.3 (only one way transmission is shown). This figure indicates that the total time when a packet is sent from computer M to computer S consists of the following several parts: times taken by the operating system (OS) of the computers, T_{Mos} and T_{Sos}; times taken by the serial ports, T_{Mport} and T_{Sport}; times taken by the wireless modules, $T_{Mmodule}$ and $T_{Smodule}$; and the transmission time from A to B, T_{AB}.

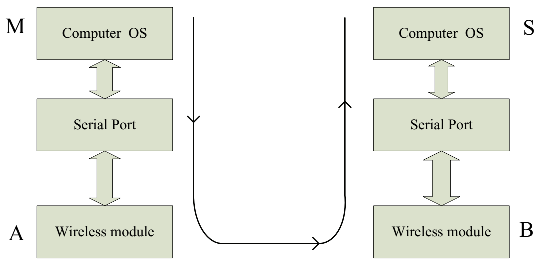


Fig. 3. Packet sending procedure

A Windows XP system is installed in each computer, and it is a multi-task operation system. A delay may happen if another task is in progress when the data request is sent out. The time taken by the OS can be divided into two parts: $T_{Mos} = T_{Mosmin} + T_{Mosvar}$, where T_{Mosmin} stands for the time that the OS handles the data request command as soon as possible,which can be seen as a constant, while T_{Mosvar} stands for the time waiting for the OS to have time to handle the data request command. Similarly, for computer S, $T_{Sos} = T_{Sosmin} + T_{Sosvar}$. For the wireless module, the time delay can also be divided into two parts: $T_{Mmodule} = T_{Mmodmin} + T_{Mmodvar}$. Module B is ready for receiving data all the time and the time $T_{Smodule}$ can be seen as a constant. In the experiment, the packet size is 25 bytes and the serial port is set with baud rate of 38400, no parity and one stop bit. So the time taken by the serial port is fixed, i.e., $T_{port} = T_{Mport} = T_{Sport} = (10 * 25)/38400 * 1000 \approx 6.5ms$. The distance between the two wireless modules is in the order of tens of meters, so the transmission time from A to B can be ignored, i.e., $T_{AB} = 0$.

When the data transmitted from B to A, the same procedure is followed. So the round trip time can be calculated as

$$\begin{aligned} T_{RD} &= 2(T_{Mos} + T_{Sos} + T_{Mport} + T_{Sport} + T_{Mmodel} \\ &\quad + T_{Smodel} + T_{AB}) \\ &= 2(T_{Mosmin} + T_{Mosvar} + T_{Sosmin} + T_{Sosvar} \\ &\quad + T_{Mmodmin} + T_{Mmodvar} + T_{Smodel} + 2T_{port}) \\ &= 2(T_{con} + T_{var} + T_{Mmodvar}) \qquad (1) \end{aligned}$$

where, $T_{con} = T_{Mosmin} + T_{Sosmin} + T_{Mmodmin} + 2T_{port} + T_{Smodel}$ and $T_{var} = T_{Mosvar} + T_{Sosvar}$. $T_{Mmodvar}$ is mainly the time waiting for the module to access the channel. T_{var} is small and will be ignored here. In this paper, we propose to use a Poisson process to approximate $T_{Mmodvar}$. In the following, we will show some experiments to verify the assertion above.

TABLE I
THE EXPERIMENT RESULTS

sets	Total Num.	Receive Ratio	Delay /(ms)		
			Min.	Max.	Ave.
(a)	36004	0.9923	36.6	52.6	39.6
(b)	43704	0.9936	36.6	85.0	39.6
(c)	27608	0.9965	36.7	60.9	39.7

Three sets of our experiments are selected and the total packet number, receive ratio, minimum delay, average delay, maximum delay of each set are depicted in Table.1. From Tab.1, we can see that the minimum delay, T_{min}, is nearly constant in the three experiments, witch corresponds to the part $2T_{con}$ in equation (1), i.e,

$$T_{min} = 2T_{con} = 36.6\ ms \qquad (2)$$

From the above equation, we obtain

$$T_{con} = 18.3\ ms$$

Subtracting T_{min} from the round trip time T_{RD}, the variable part of T_{RD} is obtained. One set of the probability density function of this part is presented in Fig.4 (the line with circle), which shows that the distribution of the varying delay is very similar to a Poisson process. And a fitting poisson distribution line is also drawn (the line with square). According to Fig.4, we can see that the variable part of the delay

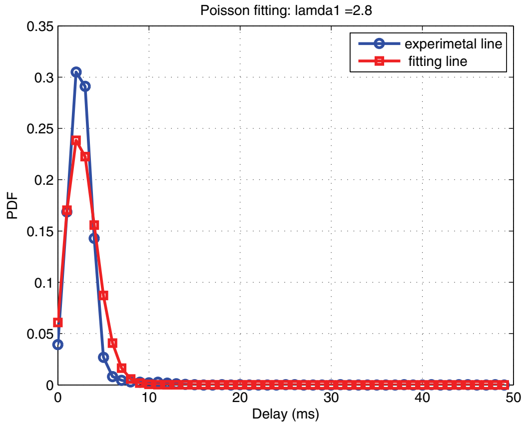
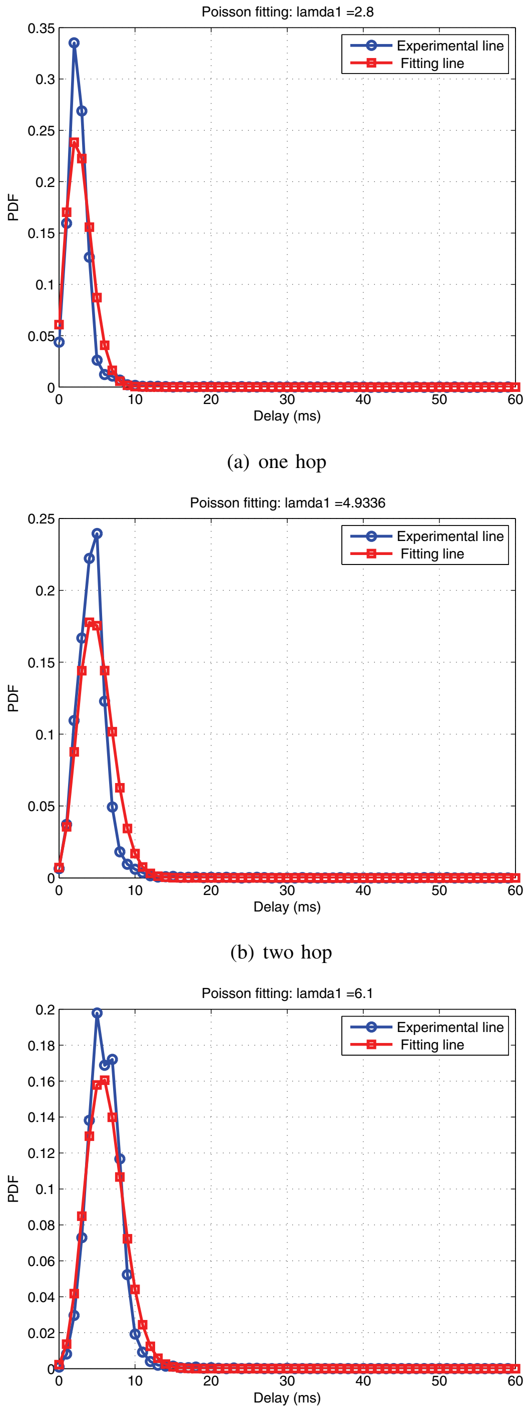


Fig. 4. PDF function of a single-hop network



(a) one hop

(b) two hop

(c) three hop

Fig. 6. PDF of a multi-hop network

can be approximated by a Poisson precess. We also see that there is some difference between the fitting line and the experiment. This may be caused by the varying delay part of the OS, i.e., T_{var}.

In Fig.5, the delay values of some packets are plotted. The red line is the delay value of the real wireless network and the blue line is the delay value simulated by the Poisson Process model using Matlab.

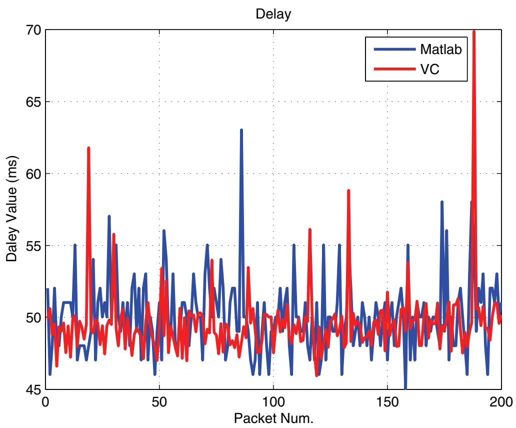


Fig. 5. PDF function of a single-hop network

B. Multi-hop Delay

Consider a a multi-hop network shown in Fig.1. Compared with the one-hop network, the only difference is the time taken by the inter nodes to retransmit the packet, denoted as $T_{retrans_i}, i = 1, \cdots, n-1$. So, the delay in the multi-hop network could be formulated as

$$T_{multiDelay} = T_{RD} + 2\sum_{i=1}^{n-1} T_{Retrans_i} \tag{3}$$

For convenience, we assume that the inter-nodes are identical. The time taken by the inter-nodes can also be divided into two parts: minimum time for the i-th node to retransmit a packet $T_{minRetrans}$, which can be seen as a constant, and time for the i-th node waiting to retransmit a packet while the channel is busy, denoted as $T_{varRetrans_i}$, which can be approximated by a Poisson process. The PDF of delays of one set of our experiments is shown in Fig.6, from which we can see that the Poisson process fits the experiment results well.

IV. State Estimation

In this section, we consider the use of the proposed time delay model in state estimation. State estimation is done by a Kalman filter[10, 11]. The purpose of this exercise is to see how good the delay model is. To this end, we will compare the state estimation errors obtained using both approaches, one with the delay model and one with the hardware setup.

Consider the following system

$$\begin{aligned} x_k &= Ax_{k-1} + w_{k-1} \\ y_k &= Cx_k + v_k \end{aligned} \tag{4}$$

where $x_k \in R^n$ is thestate vector, $y_k \in R^m$ is the measurement output, random variables $w_k \in R^n$ and $v_k \in R^m$ represent the process and measurement noise respectively. x_0 indicates the initial value of the process state. They are assumed to be independent, white, and with normal probability distributions

$$p(w) \sim N(0, Q), p(v) \sim N(0, R), p(x_0) \sim N(0, \Sigma_0) \tag{5}$$

The time update equations can be thought as *predictor* equations (6) while the measurement update equations can be thought as *corrector* equations (7) shown below.

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \tag{6a}$$

$$P_k^- = AP_{k-1}A^T + Q \tag{6b}$$

$$K_k = P_k^- C^T (CP_k^- C^T + R)^{-1} \tag{7a}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \tag{7b}$$

$$P_k = (I - K_k C)P_k^- \tag{7c}$$

The experiment includes two cases: emulation using the hardware and Matlab simulation using the Poisson process model.

In the emulation case, the linear system (4) is simulated by the master computer. The output of this system is sampled once per second and the measurements are sent to a slave computer via a wireless network. When the slave computer receives the output measurements packet, it will send back the packet back to the master computer immediately. The measured output will be recorded if the master computer receives the reply packet. And the state estimation will be done just before the next sampling time.

In the simulation case, the same linear system is simulated on Matlab. Different to the emulation part, here the wireless network is also simulated, using the Poisson process model developed before. The measured output of the system will arrive the Kalman Filter via the simulated wireless network.

We will compare the estimate errors in these two cases. Since the wireless network is introduced, there may be none or more than one packets received in each time step. So a modification needs to be made to the Kalman filter algorithm. This modification is similar to the estimation algorithm given in [12]. For convenience, we use *Kalman-predictor* to stand for the time update equations, *Kalman-corrector* for measurement update equations and *Kalman-filter* for both *Kalman-predictor* and *Kalman-corrector*. According to the packet number received in each time step, two cases are considered:(Assume the current time step is k.)

- No packet is received at time k. In this case, only the Kalman-predictor is done.
- One or more than one packets are received. In this case, firstly, consider whether the time step of the first received packet, denoted as k_0, equals the current time step k. If $k_0 = k$, do the Kalman-filtering, else, return to step k_0 and do the Kalman-filtering. Then from step $k_0 + 1$ to k, if the corresponding packet is received, do the Kalman-filtering, else, do the Kalman-predicting.

This procedure is shown in Fig.7.

We carry out the state estimation procedures using different sampling periods, i.e., 100 ms, 50 ms or 70 ms respectively and compare the traces of the estimate error covariance matrix P. Note that P is stochastic because of the time delay model. To make the comparison more meaningful, we repeat the experiment many times and compare the mean value of the trace of P.

The parameters of (4) in the experiment are chosen as follows,

$$A = \begin{bmatrix} 1.0155 & -0.0381 & 0.0022 \\ -0.0381 & 0.9965 & 0.0291 \\ 0.0022 & 0.0291 & 0.9880 \end{bmatrix}, \ C = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}^T \tag{8}$$

Figs.8–9 show the results for a one-hop network and Figs.10–11 show the results for a three-hop network. The Star is for the simulation and the line is for the emulation. We can see that they match very well.

V. Conclusions

In this paper, we have done the following: modeling and estimation of a wireless based network. Poisson process models are proposed to approximate the transmission delay of the network, both single-hop and multi-hop transmissions. Estimation is done by a Kalman filter. From the experiments, we conclude that the Poisson process model can approximate the network delays well.

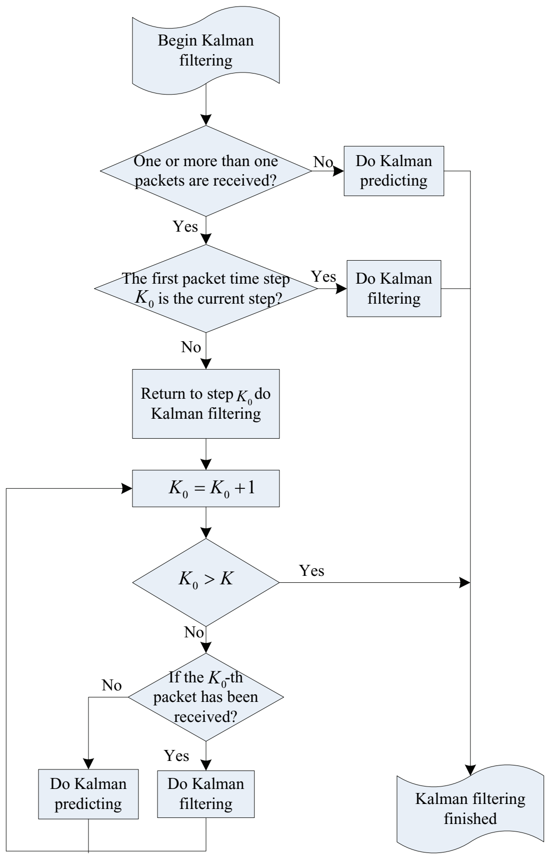


Fig. 7. Kalman filter procedure

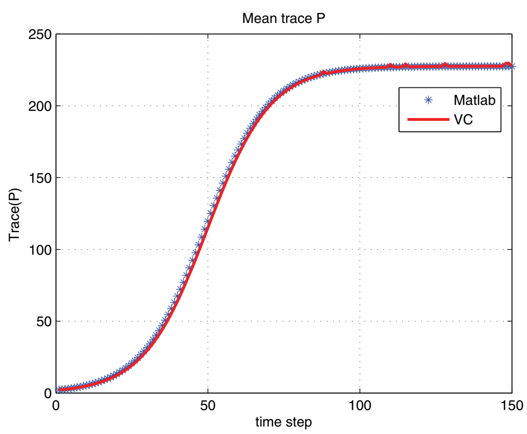


Fig. 8. Sampling period is 100 ms,single-hop

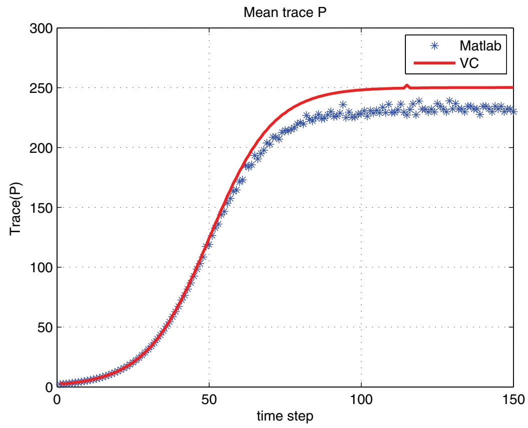


Fig. 9. Sampling period is 50 ms, single-hop

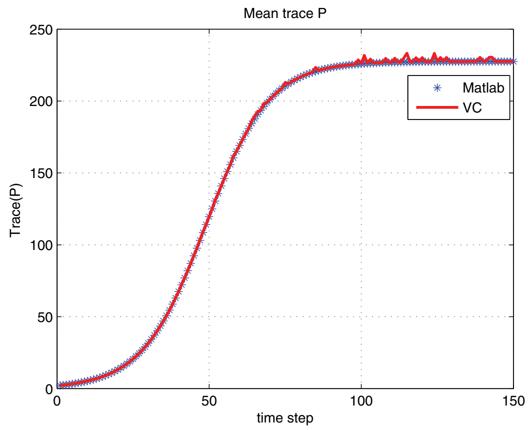


Fig. 10. Sampling period is 100 ms , multi-hop

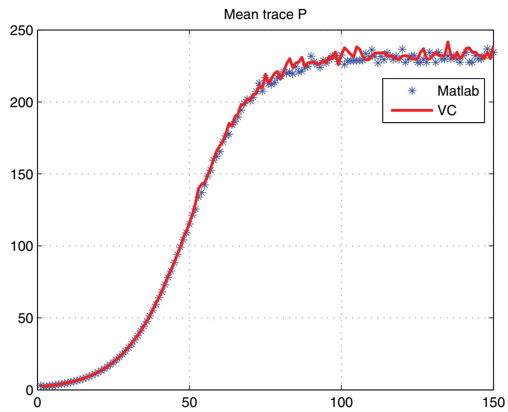


Fig. 11. Sampling period is 70 ms, multi-hop

Acknowledgment

This work is supported by The National Creative Research Groups Science Foundation of China under Grant 60721062, The National Natural Science Foundation of P.R. China under Grant 60736021, The National High Technology Research and Development Program of China under Grant 863 Program 2008AA042902 and Open Project of the State Key Laboratory of Industrial Control Technology under grant NO. ICT1002.The authors also thank Jianning Li and Tao Sun for their help in doing the experiments.

References

[1] H. Ye, G. C. Wash, and L. G. Bushnell, "Real-time mixed-traffic wireless networks," *IEEE Transaction on Industrial Electronics* , pp: 883–890, 2001.

[2] A. Willig, K. Matheus, and A. Wolisz, "Wireless technoligy in industrial networks," *Proceedings of the IEEE*, pp: 1130–1151, 2005.

[3] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine* , pp: 84–99, 2001.

[4] K. Q. Gu, V. L. Kharitonov, and J. Chen, *Stability of time-delay systems.* Brikhauser, 2003.

[5] L. Kleinrock and F. A. Tobagi, "Packet switching in radio channels: part I– carrier sense multiple-access models and their throughput-delay characteristics," *IEEE Transaction on Communications*, pp: 1400–1416, 1975.

[6] J. Misic, S. Shafi, and V. B. Misic, "Performance of a beacon enabled IEEE 802.15.4 cluster with downlink and uplink traffic," *IEEE Transaction on Parallel and Distributed Systems*, vol. 17, NO. 4, pp: 361–376, 2006.

[7] M. Martolo, S. Busanelli, and G. Ferrari, "Markov chain-based performance analysis of nultihop IEEE 802.15.4 wireless networks," *Performance Evaluation*, pp: 722–741, 2009.

[8] B. D. Noble, M. Satyanarayanan, G. T. Nguyen, and R. H. Katzy,"Trace-based mobile network emulation," *In Proceedings of. Proceedings of SIGCOMM'97*, pp: 51 61, 1997.

[9] T. R. Burchfield, S. Venkatesan, and D. Weiner, "Maximizing Throughput in ZigBee Wireless Networks through Analysis, Simulations and Implementations," *In Proceedings of the International Workshop on Localized Algorithms and Protocols for Wireless Sensor Networks Santa Fe*, pp: 15–29, 2007.

[10] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transaction of the ASME-Journal of Basic Engineering*, pp: 35–45, 1960.

[11] H. W. Sorenson, "Least-squares estimation: from Gauss to Kalman," *IEEE Spectrum*, vol. 7, pp. 63–68, 1970.

[12] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Aastry, "Kalman filtering with intermittent observations," *IEEE Transaction on Automatic Control*, vol. 49, pp: 1453–1464, 2004.