A Comparative Study on the Performance of Wireless Sensor Networks

Lin Zou Zhongwei Zhang
Department of Mathematics and Computing
University of Southern Queensland
Toowoomba, QLD, Australia
zou.usq@gmail.com zhongwei@usq.edu.au

Abstract

To increase the lifetime of the sensor networks, a recognized method is to switch off/on some sensor nodes between "sleep" and "activity" mode in order to save the energy. The performance metrics of an individual node can be calculated by using its DTMC model, the theoretical results include the average number of data units generated in a time slot λ_E, the sensor throughput T, the average buffer occupancy $\bar{B}$.

In our research, we have implemented a wireless sensor network on NS2 in which the sensors nodes can be either in sleep or activity modes. Our simulations have produced the experiment results of performance metrics. By comparing the experiment results with the theoretical results, we demonstrate the DTMC model is able to accurately describe the behavior dynamics of a sensor node.

1 Introduction

Because sensor nodes usually are so small in size that may be used conveniently in some place, where wired network can't reach or too expensive to install. The sensor nodes are powered by battery which has a limited energy resource. Furthermore, the battery is usually difficult to replace or recharge as the sensor nodes are used in some special fields. As a result, when a sensor node exhausts its power, it will stop its functions. Because of this, the network topology has been changed and the sensor network's capability including sensing phenomenon, information generation and routing data might be degraded. Many researches strive to reduce the energy consumption [3, 4, 5], etc. For instance, a research has introduced to sleep mode to the sensor node which save some energy while the sensor nodes are idle [18].

In this paper, we look at the dynamics of sensor nodes while sensor node states transit from one to another. The rest of this paper is organized as follows. The model of analyzing the performance of sensor nodes is introduced in Section 2. Simulation environment of NS2 and experiment results are shown in Section 3. We give conclusion and future work in Section 4.

2 Using Markov Chain to Represent Sensor Node Dynamics

2.1 Sensor node states

Each sensor is characterized by two operational states: activity and sleep. In activity state the node is full workload, while in sleep state it cannot take part in the network activity; thus, the network topology will be changed while nodes enter or exit the sleep state.

Based on the above observations, we describe the temporal evolution of the state of a sensor node in terms of cycles, as depicted in Figure 1. Each cycle comprises a sleep state (S) and an activity state (A).

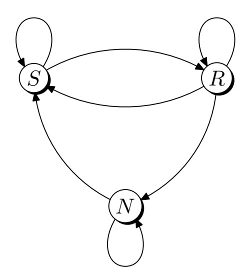

Figure 1. state evolution of a sensor node

When the sensor transmits to the activity mode, state A begins and the sensor schedules a time instant in the future at which it will go back to sleep. A node transmits its state from activity to sleep with a geometrically distributed parameter p, or keep its activity state with a parameter $(1-p)$; the node switches its state from sleep to activity with a geometrically distributed parameter q, or keeps its sleep state

6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)
0-7695-2841-4/07 $25.00 © 2007 IEEE

with parameter $(1-q)$. The scheduled periods of sleep and activity, expressed in time slots, are modeled as random variables geometrically distributed with parameter q and p, respectively.

2.2 Transition Matrix of A DTMC

The transition probabilities $T(S_d, S_o)$ is from source state S_o to destination S_d. To represent the states of the complete DTMC we use the same notation as for the simplified model, adding an superscript *W* or *F* to represent the state of the next-hops. Now, we are ready to find out the transition probabilities from one state to another. All these findings are listed in [19]. Note that each transition can be seen as going through three steps: every one is not independent of other two steps and no time order is distinct in the three steps.

To simplify the formula, we assume that $l_0 = (1-\alpha)(1-g)$, $l = \beta g + (1-\alpha-\beta)(1-g)$, $b_0 = g(1-\alpha) + \alpha(1-g)$, $b = g(1-\alpha-\beta) + \alpha(1-g)$ According calculation, we can calculate all transition probabilities of one sensor node from any state to others'. Then, we put the transition probabilities into a square matrix, which is called transition matrix.

2.3 Stationary Distribution of DTMC

Let us denote the stationary distribution of the complete DTMC by vector $\pi = \{\pi_s\}$, where *s* is one state listed above in the transition matrix. Based on Markov Chain theory, the stationary distribution vector of π can be computed. From the stationary distribution of the DTMC, we can derive,

- the average number of data units generated in a time slot, $\wedge_E$,
$$\wedge_E = \sum_{i=0}^{\infty} (\pi_{R_i^F} + \pi_{R_i^W}) \cdot g$$
- the sensor throughput *T*, defined as the average number of data units forwarded by the sensor in a time slot, is
$$T = \sum_{i=1}^{\infty} (\pi_{R_i^F} + \pi_{N_i^F}) \cdot \beta$$
- the overall probabilities π_R, π_S, π_N that a sensor is in the corresponding states *R*, *S*, *N*
- the average buffer occupancy:
$$\bar{B} = \sum_{k=1}^{\infty} [k \cdot (\pi_{R_k^F} + \pi_{R_k^W} + \pi_{N_k^F} + \pi_{N_k^W})]$$

For a particular sensor node, we just add the index as an superscript to above equations.

3 Simulation and Experiment

NS2 simulator has been around since 1989 and several institutions and societies have supported and contributed to its development. NS2 has been used to implement many famous TCP flow control algorithms and protocols, congestion control mechanisms, etc. In our simulation, we simply adopted a version of NS2 with the NRL's Sensor Network Extension [6].

3.1 Theoretic results

After given the values of α, β, w and f along with parameters p, q and g, we can derive the transition matrix. Note that the sum of each of the columns in the transition matrix equals to 1; in other word, that this must be true since a node must be in one of the states after one step transition. The transition matrix has been proved to be stable after a certain number of steps.

For example, let's assume that $\alpha = 0.05$, $\beta = 0.05$, $p = q = 0.1$, $g = 0.005$, are given, then the transition matrix is stable after 640 steps.

Now, we assume that at the beginning of each trail run, all sensor nodes are in sleep state without any data in the network that means all nodes are in state S_0^W, although the stationary distribution is not affected by initial probabilities vector:

For instance, all sensor nodes eventually come to a steady state, and the distributions are:
$$\begin{aligned} q^{640} &= T^{640} \cdot q \\ &= [0.1382, 0.1121, 0.0896, 0.0804, 0.0267, \\ &\quad 0.0277, 0.1812, 0.1592, 0.0085, 0.0092, \\ &\quad 0.0592, 0.0518, 0.0028, 0.0031, 0.0193, \\ &\quad 0.0168, 0.0012, 0.0014, 0.0064, 0.0053]^T \end{aligned}$$

Once we know the stable distribution, we can compute the metrics shown in Section 2, $\wedge_E$, T, and $\bar{B}$.
$$\begin{aligned} \wedge_E &= \textstyle\sum_{i=0}^{\infty} (\pi_{R_i^F} + \pi_{R_i^W}) \cdot g \\ &= [(0.08961+0.08044)+(0.02666+0.02765)+ \\ &\quad (0.00845+0.00923)+(0.00278+0.00307)+ \\ &\quad (0.00121+0.00142)] \times 0.005 \\ &= 0.25051 \times 0.005 \\ &= 0.00125282 \end{aligned}$$
and
$$\begin{aligned} \mathrm{T} &= \textstyle\sum_{i=1}^{\infty} (\pi_{R_i^F} + \pi_{N_i^F}) \cdot \beta \\ &= [(0.02666 + 0.15923) + (0.00845 + \\ &\quad 0.05179) + (0.00278 + 0.01682) + \\ &\quad (0.00121 + 0.00525)] \cdot 0.05 \\ &= 0.27219 \times 0.05 \\ &= 0.0136095 \end{aligned}$$
and

6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)
0-7695-2841-4/07 $25.00 © 2007 IEEE

$$\begin{aligned}\bar{B} &= \sum_{k=1}^{\infty}[k \cdot (\pi_{R_k^F} + \pi_{R_k^W} + \pi_{N_k^F} + \pi_{N_k^W})] \\ &= [1 \times (0.02666 + 0.02765 + 0.18120+ \\ &\quad 0.15923) + 2 \times (0.00845 + 0.00923+ \\ &\quad 0.05918 + 0.05179) + 3 \times (0.00278+ \\ &\quad 0.00307 + 0.01928 + 0.01682)+ \\ &\quad 4 \times (0.00121 + 0.00142+ \\ &\quad 0.00641 + 0.00525)] \\ &= 0.39474 + 0.2573 + 0.12585 + 0.05716 \\ &= 0.83505\end{aligned}$$

In next section, we compare the theoretic metrics with the experiment results gained from NS2 simulation.

3.2 Comparison of Result Between Simulation and Calculation

Two experiment scenarios, having 25 nodes and 100 nodes, are tested. In each of scenarios, the nodes are spread around in a square and the sink is away from the square. The detailed configuration parameters are shown in Table 1.

Parameters	scenario 1	scenario 2
The number of nodes	25	100
The length of a time slot	0.1 s	
Simulation times	120, 180 and 240 s	
maximum radio range: r	0.25	
Reception in a time slot: α	0.05	
Transmission in a time slot: β	0.05	

Table 1. Parameters for two experiment scenarios

- Results comparison of scenario 1 where the simulation time is 120 seconds

  Figure 2 shows that the simulation values of the average data generation are not close to the modeling values even they are incompact to distribute around the diagonal $y = x$. The difference is approximately in between from -4% to 3%. Figure 3 shows that the simulation values are away from the modeling values, which the difference is approximately in between from -4.8% to 4.5%. The distribution of throughput values is incompact on the Figure 6. Figure 4 shows the average buffer occupancies of sensor nodes also incompactly distribution on the figure. The difference of simulation and modeling values is approximate between from -4% to 4%. The result is not close to modeling values.

- Results comparison of scenario 1 where the simulation length is 180 seconds

  Figure 5 shows that the simulation values are closer to the modelling values than the simulation in 120 seconds, which the difference is approximately in between from -1% to 1.5%. Figure 6 shows that the simulation values are closer to the modelling values than the simulation in 120 seconds, which the difference is approximately in between from -1.2% to 1.2%. Figure 7 shows the difference of simulation and modeling values is approximate between from -1.6% to 1.6%. The result is also closer than simulation in 120 seconds. But it is hardly to present where the sensor nodes distribute on the figure.

- Results comparison of scenario 1 where the simulation length is 240 seconds

  Figure 8 shows that the simulation values are quite close to the modeling values, which the difference is approximate between from -1% to 1.5%. We notice that the results of running in 240 seconds is similar with running in 180 seconds Figure 9 shows that the simulation values are close to the modelling values, which the difference is approximate between from -1% to 1%. Figure 10 shows that the simulation values are close to the modelling values. We can see that the difference of simulation and modeling values is approximate between from -1% to 1.5%.

  In summary, the simulating values in 120 seconds running and the modeling values are not quite approach each other. Whereas in 180 and 240 seconds running, the simulating values are closer to modeling values than 120 seconds. This result validates that our simulation need a long time running, almost 180 or 240 seconds, to match the model.

  However, the result does not have persuasion for few nodes in a sensor network. So we increase the number of nodes in a sensor network to 100 nodes to compare the difference between two scenarios still in terms of three running time.

- Results comparison of scenario 2 where the simulation length is 120 seconds

  Figure 11 shows that the simulation values are not very close to the modelling values, which the difference is approximate between from -4% to 3%. The average generation rate of sensor nodes mainly concentrate on 0.000124 to 0.000144 that means the average generation rate of most nodes are between 0.000124 and 0.000144. It also represents that the sensor network running in 120 seconds does not reach stable. Figure 12 shows that the simulation values are away from the modelling values, which the difference is approximate between from -4% to 4.5%. The distribution of throughput values is incompact on the Figure 12. Figure 13 shows the average buffer occupancies of sensor nodes distribute uniformly on the figure.

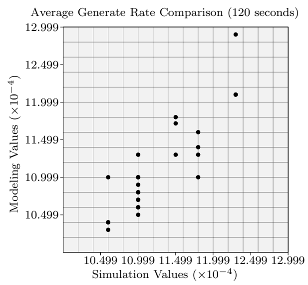

Figure 2. Average generation rates

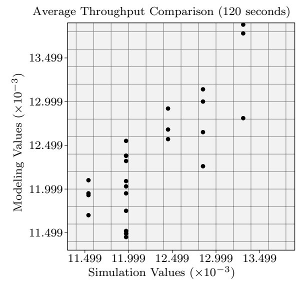

Figure 3. Throughput

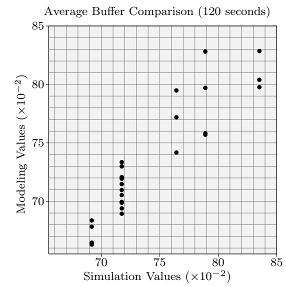

Figure 4. Average buffer occupancy

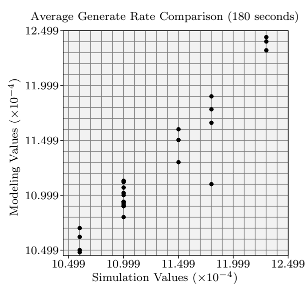

Figure 5. average generation rates

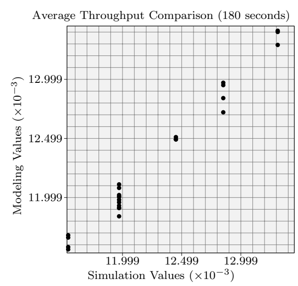

Figure 6. Throughput

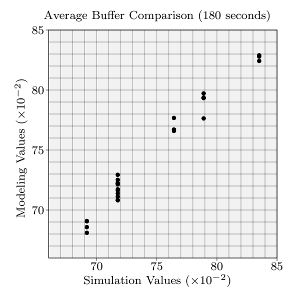

Figure 7. The average buffer occupancy

This means that the buffer of each node stored the difference number of data units. We can see that the difference of simulation and modeling values is approximate between from -4% to 4%. The result is also not close to modeling values.

- Results comparison of scenario 2 where the simulation length is 180 seconds

  Figure 14 shows that the simulation values are closer to the modelling values than the simulation in 120 seconds, which the difference is approximate between from -1.6% to 1.7%. The average generation rate of sensor nodes mainly concentrate on 0.000122 to 0.000142. It also represents that the sensor network running in 120 seconds does not reach stable. Figure 15 shows that the simulation values are closer to the modelling values than the simulation in 120 seconds, which the difference is approximate between from -1.6% to 1.7%. The throughput of sensor nodes mainly concentrate on 0.00132 to 0.00152. Figure 16 shows the average buffer occupancies of sensor nodes distribute uniformly on the figure. This means that the buffer of each node stored the difference number of data units. We can see that the difference of simulation and modeling values is approximate between from -1.6% to 1.7%. The result is also closer than simulation in 120 seconds.

- Results comparison of scenario 2 where the simulation length is 240 seconds

  The average generation rates of sensor nodes mainly concentrate on 0.000122 to 0.00014. Figure 17 shows that the simulation values are quite close to the modelling values, which the difference is approximate between from -1% to 1.1%. The throughput of sensor nodes mainly concentrate on 0.00132 to 0.0015. Figure 18 shows that the simulation values are close to the modelling values, which the difference is approximate between from -1.1% to 1.1%. Figure 19 shows the average buffer occupancies of sensor nodes distribute uniformly on the figure. This means that the buffer of each node stored the difference number of data units. We can see that the difference of simulation and modeling values is approximate between from -1.1% to 1.1%.

3.3 Comparison summary

In summary, the simulating values in 100 nodes scenario is much more accurate to present the nodes ’ behaviors than 25 nodes scenario in terms of average generation rate, throughput and buffer occupancy. The simulation is easy to modify the number of nodes in a sensor network to validate the result.

Furthermore, the results show that the difference between 120 and 180 seconds are large; but they are quite similar between 180 and 240 seconds. This means that after

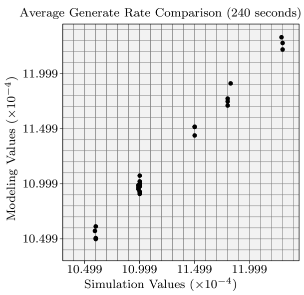

Figure 8. average generation rates

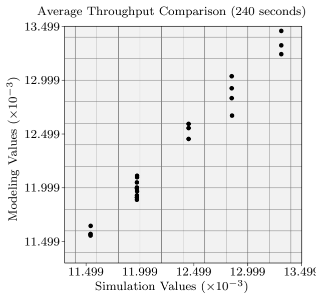

Figure 9. Throughput

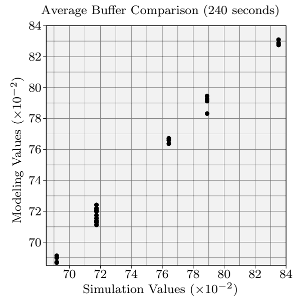

Figure 10. The average buffer occupancy

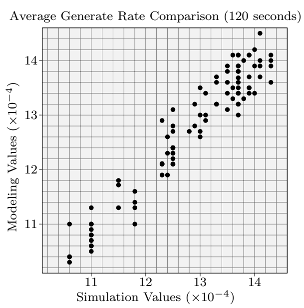

Figure 11. average generation rates

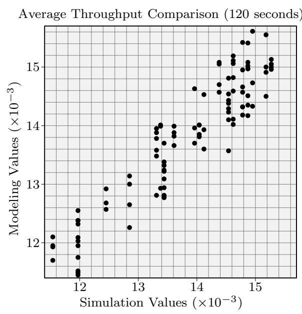

Figure 12. Throughput

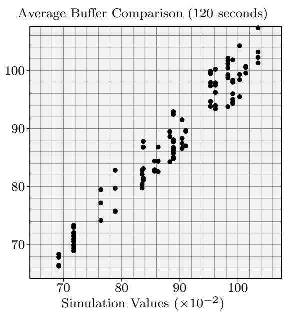

Figure 13. The average buffer occupancy

180 seconds running the network reaches stable.

4 Conclusion

In this paper, we described in great details the approach of switching the sensor nodes between“sleep” and “activity” states based on the Geometric distribution. We also demonstrated how to use the Markov chain to model the dynamics of sensor nodes. The DTMC based analysis is used to calculate the sensor node's performance metrics including the throughput, the average generation rate and the average buffer occupancy. To validate the DTMC based analytic model, we run a large number of simulations for two scenarios.

The results showed that after a long time running, the simulation and modeling metrics are very close in terms of the average generation rate, throughput and average buffer occupancy. Moreover, through simulation we noticed that the sensor network become stable after 180 seconds running.

In the future, we are going to extend the simulations and experiments to the general cases. We will investigate the performance of the sensor network, in which the changes of network architectures and topology may be studied further.

References

[1] Fan Ye, Haiyun Luo, Jerry Cheng, Songwu Lu, Lixia Zhang. *A Two-Tier Data Dissemination Model for Large-scale Wireless Sensor Networks*.MOBICOM02, September 2328, 2002, Atlanta, Georgia, USA.

[2] C.-F. Chiasserini and M. Garetto. *Modeling the Performance of Wireless Sensor Networks*. IEEE INFOCOM 2004.

[3] Qiangfeng Jiang, D. Manivannan. *Routing Protocols for Sensor Networks*.

[4] G. Asada, M. Dong, T. S. Lin, F. Newberg, G. Pottie, W. J. Kaiser. *Wireless Integrated Network Sensors: Low Power Systems on a Chip*.

[5] Wei Ye, John Heidemann, Deborah Estrin. *An Energy-Efficient MAC Protocol for Wireless Sensor Networks*. The Proceedings of the IEEE INFOCOM, 2002.

[6] Ian Downard, Naval Research Laboratory.*SIMULATING SENSOR NETWORKS IN NS-2*.NRL Formal Report 5522-04-10, 2004.

[7] Antonio G. Ruzzelli, Gregory OHare, Raja Jurdaky, and Richard Tynan. *Advantages of Dual Channel*

0-7695-2841-4/07 $25.00 © 2007 **IEEE**

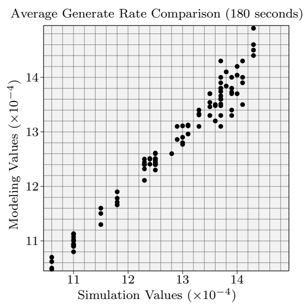

Figure 14. average generation rates

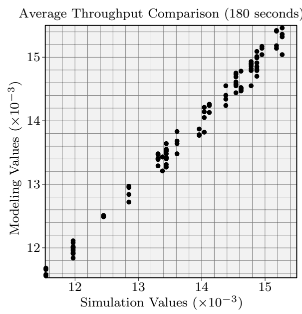

Figure 15. Throughput

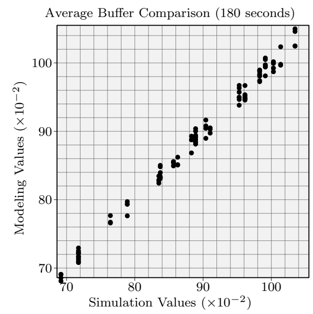

Figure 16. The average buffer occupancy

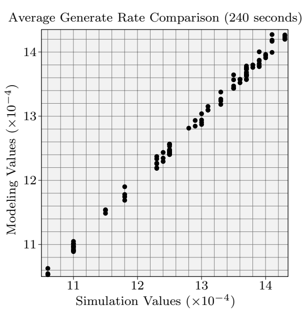

Figure 17. average generation rates

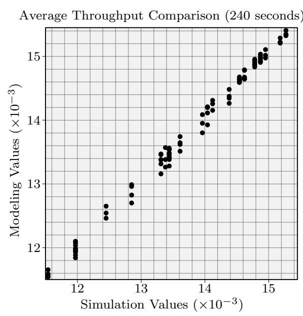

Figure 18. Throughput

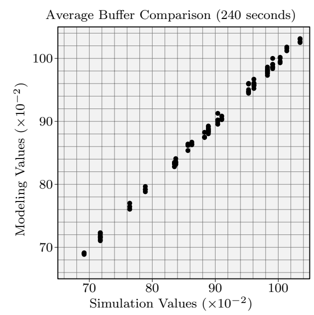

Figure 19. The average buffer occupancy

MAC for Wireless Sensor Networks. In proceedings of the 1st International Workshop on Software for Sensor Networks (SensorWare) at COMSWARE06. New Delhi, India. January, 2006

[8] Bret Hull, Kyle Jamieson, Hari Balakrishnan. *Mitigating Congestion in Wireless Sensor Networks*. SenSys'04, November 3.5, 2004, Baltimore, Maryland, USA.

[9] Chieh-Yih Wan, Andrew T. Campbell, Lakshman Krishnamurthy. *PSFQ: A Reliable Transport Protocol for Wireless Sensor Networks*.WSNA02, September 28, 2002, Atlanta, Georgia, USA.

[10] Rebecca Atherton, Leslie Hogben. *A Look at Markov Chains and Their Use in Google*. Iowa State University, MSM Creative Component, Summer 2005.

[11] Hogben, Leslie. *Elementary Linear Algebra*. West Publishing Company, St. Paul, MN, 1987: 81-92

[12] Hefferon, Jim. (2003). *Linear Algebra*, chapter 3. Retrieved 4/20/05 from

[13] National Council of Teachers of Mathematics (NCTM). *New Topics for Secondary School Mathematics: Matrices*. Reson, VA: NCTM, 1988: 1, 2, 49-65, 96-98.

[14] Carter, Tamara; Tapia, Richard A.; Papakonstantinou, Anne. (1995-2003). *Linear Algebra: An Introduction to Linear Algebra for Pre-Calculus Students*, chapter 7, Retrieved 4/20/05 from http://ceee.rice.edu/Books/LA/index.html

[15] W. Rabiner Heinzelman, A. Chandrakasan, and H. Balakrishnan, *Energy-Efficient Communication Protocol for Wireless Microsensor Networks*. 33rd International Conference on System Sciences (HICSS 00), Jan. 2000.

[16] A. Ephremides, *Energy Concerns in Wireless Networks* IEEE Wireless Communications, Aug. 2002.

[17] K. Fall and K. Varadhan. *The NS Manual (Formerly NS Notes and Documentation*. http://www.isi.edu/, 2002.

[18] Wei Ye and John Heidemann. *Medium Access Control in Wireless Sensor Networks*. USC/ISI TECHNICAL REPORT ISI-TR-580, OCTOBER 2003.

[19] C.-F. Chiasserini and M. Garetto. *Modeling the Performance of Wireless Sensor Networks*. IEEE INFOCOM 2004.