Energy Conservation for Multi Transitioned Wireless Sensor Networks

Mznah Al-Rodhaan
Computer Science Department, College of Computer and Information Sciences
King Saud University,SAUDI ARABIA
rodhaan@ksu.edu.sa

Abstract

Wireless sensor networks are battery operated so its energy should be conserved intelligently. Since part of this energy is consumed during state transition, the number of transitions should be reduced. Our schema uses time-controlled queues to reduce the number of transitions. An analytical model is developed to analyze the performance of the proposed scheme in terms of energy consumption and end-to-end delay. The performance study shows that the average energy consumption is reduced after using time-controlled queues. Queuing will reduce the energy consumption but with a trade-off in end-to-end delay. A time-controlled queue is used to reduce end-to-end delay compared to threshold-controlled queues.

Keywords: *Wireless Sensor Network; HSNs; Power Conservation; End-to-end Delay; Controlled queue.*

1. Introduction

Nodes in wireless network typically have good portability and flexibility. However, in many cases they are equipped with limited capacity power which makes them heavily constrained by battery lifetime. In some scenarios, a node may exhaust its power supply where a replacement of power resources might be impossible. In many applications therefore, power conservation is a key aim; however, increasing the power dedicated to radio transmission and reception can broaden the radio range improving connectivity and boosting network functionality. Clearly, there is often a trade-off between the connectivity needed and the amount of energy consumed. Researchers have put considerable effort into the design of power-aware protocols [1]. A survey of energy efficient protocols for sensor networks could be found in [2] where the authors concluded that the best method for energy conservation is to turn off inactive sensors without effecting the functionality of the system.

In this paper, we are proposing a new energy optimization scheme by which the average energy consumption of individual nodes in the sensor network is reduced during packet transmission based on queue threshold within a certain time period. An analytical model is developed for a cluster-based sensor network by analyzing the system performance in terms of average energy consumption and delay. The rest of this paper is organized as follows: in section 2, related work is discussed. The system model is presented in section 3. In section 4, we present the performance analysis and provide analytical study for determining the end-to-end delay and power consumption of the powerful sensor node. In section 5, results and discussion are presented. Section 6 concludes the study.

2. Related Work

The dual-state model was presented as an analytical model in [3] to study the system performance in terms of energy consumption, network capacity, and mean delay. The sensor dynamically switches between active and sleep modes where there is a trade-off between energy consumption and mean delay. Dual-modes [4] are used to minimize power consumption in sensor networks. A new energy efficient routing protocol was proposed where inactive sensors switch to a sleeping state for energy conservation. The aim of this method was to preserve the lifetime of sensor networks by using active and sleep modes.

Most of the time sensor networks are considered to be homogeneous networks assuming that all sensor nodes have the same capabilities such as computation power, memory storage, power supply, communication capabilities...etc. To improve end-to-end delay in data delivery when using multi-hop communication, clusters are formed in the sensor networks then a cluster-head is selected for each cluster. Therefore, all nodes in a cluster will communicate with the cluster-head directly but with the risk of consuming a considerable amount of energy at the initialization phase. Heterogeneous Sensor Networks (HSNs) consist of different types of sensor nodes. HSNs have been studied in the literature [5, 6] because HSNs can significantly improve sensor network performance [5]. To improve energy consumption in WSN, a new energy minimization technique was introduced [7] using IDLE and BUSY states where minimizing the energy consumed depends on queue threshold regardless of the arrival rate.

3. System Model

In cluster network, nodes are divided into non overlapping clusters grouped according to certain criteria, where each node is assigned different roles. There are cluster-heads gateways and ordinary nodes. A cluster-head manages its own cluster, coordinates intra/inter-cluster communication and so on, a gateway establishes the communication between different clusters, while an ordinary node belongs to a certain cluster and communicates with their cluster-head [8].

We will consider a heterogeneous sensor network clustered model, which consists of two physically different types of sensors. A small number of powerful sensors and a large number of ordinary sensors uniformly distributed in the field. The powerful sensors consist of cluster-heads and gateways while the ordinary sensors represent nodes within a cluster. Each cluster consists of a number of ordinary nodes and at least two powerful nodes; one as cluster-head and the rest as gateways. After deployment, clusters are formed and a powerful node in each cluster serves as the cluster head. Figure 1 shows the cluster formation in this clustered model. Powerful nodes have longer transmission range, higher data rate and more power supply than ordinary nodes.

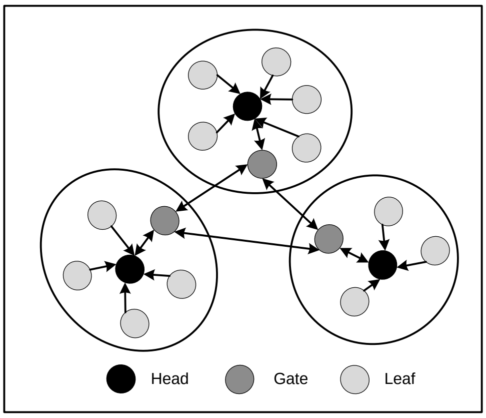

Figure 1: The clustered architecture

Routing in the Heterogeneous Sensor Network (HSN) is basic where each node sends data to its cluster head. Then the cluster-head collects data from multiple nodes within its cluster and sends the collected data to the Base Station (BS) with the help of other powerful nodes. The two major states of a sensor node are sleep state and active state [4] as in Figure 2. The sleep state corresponds to a very low value of energy consumption; when asleep, a node cannot communicate with the external world. On the other hand, during the active state a sensor node will communicate with others to transmit and receive data packets. Before transmitting data the node needs to prepare the data packets and search for a route. This preparation time differs from sender and intermediate nodes.

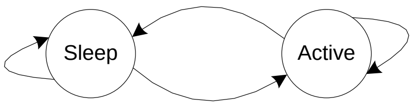

Figure 2: The two major states of a sensor node.

For the powerful sensors, the active state should be divided to three sub-states with different levels of energy consumption. A node needs to be in one of three modes: listening, sending or receiving for sensing, communicating and data processing respectively. Each of these modes defer in the amount of energy consumed. In the listening mode, the sensor node listens to the wireless channel during the contention period where the sensing consumes less energy than sending or receiving. In the receiving mode, energy is consumed mainly by the transceiver electronics such as demodulation and decoding which is more than the consumption of the data processing unit. In the sending mode, signals need processing including modulation or coding plus amplification. Thus, the sending mode consumes more power than the two other modes. In our proposed schema, the active state is divided to three sub-states as modes namely *listen*, *send* and *receive* modes. Therefore, we will model the active state as three sub states as shown in Figure 3.

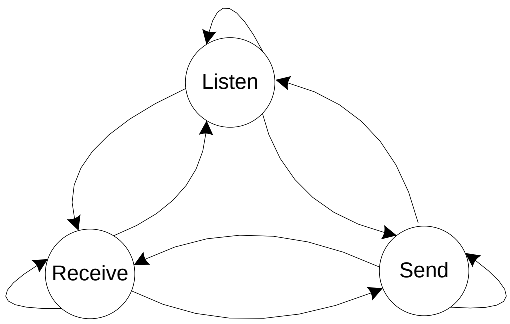

Figure 3: The active state divided into three sub states.

Cluster heads and gateways can be in any three sub states during their period of active states. The powerful node will switch to *send* mode when its buffer exceeds the queue threshold (number of packets = *N)* or after *T* time if the buffer is not empty to be fair in case of bursts of fewer messages as in Figure 4. In a low arrival rate, the few messages will suffer a longer delay than in the case of higher arrival rate unless the process of queuing is controlled by time. The de-queuing process will be triggered by reaching the first of two events: reaching the queuing threshold or time threshold. The node switches between *send* mode and *listen* mode until the communication session is done. For those nodes, most of the energy is consumed during transition between different modes so the average energy consumption depends on the chosen buffer threshold and time limit. It is very important to carefully choose the threshold value for which the Heads and Gates nodes consume less energy but without increasing the end-to-end delay between sender and receiver. Moreover, waking up Gates and other Heads consume more energy due to transition between *active* and *sleep* states. Our algorithm minimizes the power consumption by avoiding frequent transmissions of individual messages.

Steps preformed by a powerful node upon receiving a Message s
y = previous number of messages.

```
In-queue(Q,s)
y = y+1
If (y>= N or T >= time )
        while (Q)
                x = de-queue(Q)
                transmit (x)
        End while.
End if.
```

Figure 4: Steps performed by the powerful node to control power consumption.

4. Performance Analysis

We are focusing on minimizing the energy consumption of individual powerful nodes, Heads and Gates, during their period of *active* state based on the choosing threshold; so we will try to analyze the behavior of a powerful sensor node. In the proposed scheme, identical powerful nodes and identical ordinary nodes are assumed.

A sensor network can be modeled analytically as a network of queuing systems [7] because the powerful nodes receive messages of different sizes such as data or control packet, queue them until the threshold is reached then transmit them. The whole system can be modeled as a network of queuing systems operating in steady state where each powerful sensor is a queuing system by itself. For simplicity, two assumptions were made: 1) Packet generation and arrival at each node assumed to be independent and identically distributed. 2) Each node has infinite buffers to avoid dropped packets. So each powerful node is modeled as M/G/1 system [9] that satisfies the following conditions: 1) service delays are independent and have a general distribution because packets differ in size. 2) Packets arrive at each node according to a Poisson process with rate λ and independent of service time. 3) Packets delivered from gateway to the BS have mean service time $(1/\mu)$ and the channel access time follows general distribution with mean $1/\gamma$. Each system has a single server that serves packets in their order of arrival (FCFS). When the packet is ready to be transmitted, the node will follow the used protocol at the MAC layer, so this contention time is included in the service time. Powerful nodes are modeled as M/G/1 systems with different arriving customers such as messages. The arrival of data packets follows a Poisson process with mean arrival rate per node (λ). During their period of active time, powerful nodes in a cluster remain in *listen* state and switches to *send* state when the node's buffer is filled at least with threshold number of packets (N) or after T time and switches back to *listen* state when there are no packets in the node's buffer. We analyze the performance of the system in terms of the following parameters.

4.1 Queuing Analysis

All packets are subject to different amounts of delay while travelling from source to destination in any network such as queuing delay, processing delay, and propagation delay...etc. These delays depend on many factors such as: energy level, packet length, and contention level at that particular time. Propagation delay is assumed to be negligible in this analysis since packets in wireless communications travels by the speed of light where propagation delays= $\text{distance}/(3 * 10^8)$. However, other delays affect the network performance. A queuing delay experienced by the packets at any powerful node is defined as the average waiting time of the packets within a queue. The end-to-end delay will be affected since one of its components is the queuing delay. Based on M/G/1 queuing system, the mean number of packets in the queue is (N_Q). Table 1 lists the parameters used in the queuing system.

Table 1: Parameters of the Queuing Network Model

N_Q	Average number of packets waiting in the queue
$\mathbf{N_i}$	number of packets waiting in the queue when the i^{th} packet arrives
$\boldsymbol{N}$	The queue threshold
$\boldsymbol{\lambda}$	Arrival rate
$\boldsymbol{\mu}$	Service rate
$\mathbf{1}/\boldsymbol{\gamma}$	Average channel access time
$\boldsymbol{\rho}$	Utilization factor of the server $(\rho < 1)$
W	Average packet waiting time in the queue
$\mathbf{W_Q}$	Waiting time for N_Q.
$\boldsymbol{S}$	Average service time per packet
$\mathbf{w_i}$	Waiting time for the i^{th} packet in the queue
R	Average residual service time
$\mathbf{r_i}$	Residual service time
X	Average server service time
$\mathbf{x_i}$	Server service time for the i^{th} packet

The service time for any packet $(\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots)$ is a discrete random variable where the average service time $\overline{\boldsymbol{X}} = \mathbf{1}/\boldsymbol{\mu}$ and $\boldsymbol{E}\{\boldsymbol{X}\} = \overline{\boldsymbol{X}}$. So the average waiting time in the queue for the i^{th} route request $(\boldsymbol{w_i})$ is consisting of service times $(\boldsymbol{x_j})$ of the packets currently waiting in the queue, residual time $(\boldsymbol{r_i})$, plus the waiting time for the average number of packets in the queue. Residual service time is the remaining time of the packet currently in service when the i^{th} packet arrived.

$$\boldsymbol{w_i} = \sum_{j=i-N_i}^{i-1} \boldsymbol{x_j} + \boldsymbol{r_i} + \mathbf{W_Q} \qquad (\mathbf{4.1})$$

$$\boldsymbol{E}\{\boldsymbol{w_i}\} = \boldsymbol{E}\left\{\sum_{j=i-N_i}^{i-1} \boldsymbol{E}\left\{\boldsymbol{x_j} \middle| \boldsymbol{N_i}\right\}\right\} + \boldsymbol{E}\{\boldsymbol{r_i}\} + \boldsymbol{E}\{\mathbf{W_Q}\} \qquad (\mathbf{4.2})$$

Knowing that $\boldsymbol{N_i}$ is a random variable and independent of $\boldsymbol{x_j}$.

$$\boldsymbol{E}\{\boldsymbol{w_i}\} = \overline{\boldsymbol{X}}\boldsymbol{E}\{\boldsymbol{N_i}\} + \boldsymbol{E}\{\boldsymbol{r_i}\} + \boldsymbol{E}\{\mathbf{W_Q}\} \qquad (\mathbf{4.3})$$

Following the analysis in [9] where all long-term averages viewed as limits when packet index converges to infinity, assuming these limits exist. This assumption is true if $\boldsymbol{\rho < 1}$. In other words, the arrival rate $(\boldsymbol{\lambda}) <$ the service rate $(\boldsymbol{\mu})$ so the node can handle the packet received in reasonable time and avoid the unpleasant effect of saturation [10].

$$\lim_{i\to\infty} E\{w_i\} = \bar{X}\lim_{i\to\infty} E\{N_i\} + \lim_{i\to\infty} E\{r_i\} + \lim_{i\to\infty} E\{\mathrm{W_Q}\} \qquad (4.4)$$

$$\mathrm{W} = \bar{X}N_Q + R + \mathrm{W_Q} \qquad (4.5)$$

Applying Little's Theorem as in [9]

$$N_Q = \lambda \mathrm{W} \qquad (4.6)$$

Substituting equation (4.6) in (4.5) and using $\boldsymbol{\rho = \bar{X}\lambda}$:

$$\mathrm{W} = \rho \mathrm{W} + R + \mathrm{W_Q} \qquad (4.7)$$

$$\mathrm{W} = \frac{R + \mathrm{W_Q}}{(1 - \rho)} \qquad (4.8)$$

Where the average residual time as stated in [9] is:

$$R = \frac{\lambda\ \overline{x^2}}{2} \qquad (4.9)$$

The second moment $(\overline{\boldsymbol{x^2}})$ of service time is computed as in [11]:

$$E\{X^2\} = \sum_{x_i} P(x_i)\, {x_i}^2$$

The average of waiting time formula can be obtained similar to [9, 10] by substituting (4.9) into (4.8):

$$\mathrm{W} = \frac{\lambda\overline{x^2} + 2\mathrm{W_Q}}{2\ (1-\rho)} \qquad (4.10)$$

Total service time for one packet $(\boldsymbol{S})$ in all kind of sensors can be obtained by adding the waiting time in the queue $(\boldsymbol{W})$ and the average server service time $(\boldsymbol{X})$ that include the waiting time for the channel to be free.

$$S = W + X \qquad (4.11)$$

The end-to-end delay $(\boldsymbol{D})$ is the total service time $\boldsymbol{S}$ plus the transmission delay $(\boldsymbol{T_x})$

$$\boldsymbol{D = W + X + T_x} \qquad (\mathbf{4.12})$$

The total service time $(\boldsymbol{Total_S})$ for *N* packets processed by a powerful sensor can be obtained by:

$$\boldsymbol{Total_S(T)} = \begin{cases} \boldsymbol{S * N}, & \boldsymbol{T < t} \\ \boldsymbol{S * i}, & \boldsymbol{T \geq t} \end{cases} \qquad (\mathbf{4.13})$$

While the end-to-end delay $(\boldsymbol{D_S})$ for those *N* packets can be obtained by:

$$\boldsymbol{D_S(T)} = \begin{cases} \boldsymbol{D * N}, & \boldsymbol{T < t} \\ \boldsymbol{D * i}, & \boldsymbol{T \geq t} \end{cases} \qquad (4.14)$$

Equations (4.13) and (4.14) are considering which event is happened first; the queue threshold or time exceeded *t* time.

4.2 Energy Consumption

During the cycle of active time, each powerful node waits during the *listen* state for the number of packets to reach the queue threshold N or for T time whichever comes first. If threshold value is reached due to the arrival of N packets, the powerful sensor node should wait for the channel to be free for a mean channel access time 1/γ [12]. When the channel is available, the node switches to Send state and start transmitting. For synchronization purposes with the BS, a preamble packet should be sent first [13] then the powerful sensor will be ready to transmit all packets found in the buffer to the BS then switches back to the *listen* state to be ready for another cycle. Most of the energy is consumed during transmission or state transition. Selecting a specific threshold affects the performance of the whole network because it might increase the number of cycles within a time interval. The focus here is to reduce the average energy consumption during transmission based on queue threshold.
The first order radio model is commonly used energy consumption model [14]. This model was given by:

$$E_{Tx}(k,d) = (E_{elec} + E_{amp}) \times k \times d^{\alpha} \qquad (4.15)$$

$$E_{Rx}(k) = E_{elec} \times k \qquad (4.16)$$

where $E_{Tx}(k,d)$ is the energy consumed by the transmitter to send a k-bit long packet over distance d, $E_{Rx}(k)$ is the energy consumed by the receiver when receiving a k-bit long packet, $E_{elec}(k)$ is the energy used by the electronics of the transmitter or the receiver, and E_{amp} is the energy boosted by the transmitter amplifier. Typical theoretical values: $E_{elec} = 50\ nJ/bit$, $E_{amp} = 100 \frac{pJ}{bit}/m^2$, and a path loss exponent α = 2 for a distance less than some crossover value.

It is clear that receiving cost less energy than transmitting but receiving is not a low cost operation. Consequently, any proposed protocols should consider not only the number of transmitting messages but also number of receiving messages as well as number of transitions.

During Active state, most of energy consumed results from data transmission, state transition, and reception of data. So we need to consider $E_T(no)$ which is the energy consumed due to state transitions along with $E_{Tx}(k,d)$ and $E_{Rx}(k)$.

$$E_T(no) = E_{elec} \times no \qquad (4.17)$$

Powerful nodes work as a connecting point between other sensors within the cluster or with other powerful nodes in neighboring clusters. So these nodes always pass messages by receiving then transmitting. If we have N messages, a cluster environment without Active state will consume:

$$E_{Total}(N) = N(E_{Tx}(k,d) + E_{Rx}(k)) + E_T(4N) \qquad (4.18)$$

Each message received needs two other transitions from *listen* to *receive* then to *listen* again due to shared media. Moreover, two other transitions happened when sending a message without queuing resulting in four transitions per message. Since we are having N messages, we will have $E_T(4N)$.

Queuing messages then sending them will reduce the number of transitions to 2N + 2.

$$E_{Total}(N) = N(E_{Tx}(k,d) + E_{Rx}(k)) + E_T(2 + 2N) \qquad (4.19)$$

5. Discussion

End-to-end delay is affected by the total service time. The total service time increases linearly as the queuing threshold (N) increases when queuing threshold is the only factor controlling the number of cycles. The main problem here occurs with larger thresholds or low arrival rate where all packets have to wait for longer time. In our algorithm, the delay is controlled better when the queuing threshold is overruled with the time threshold.

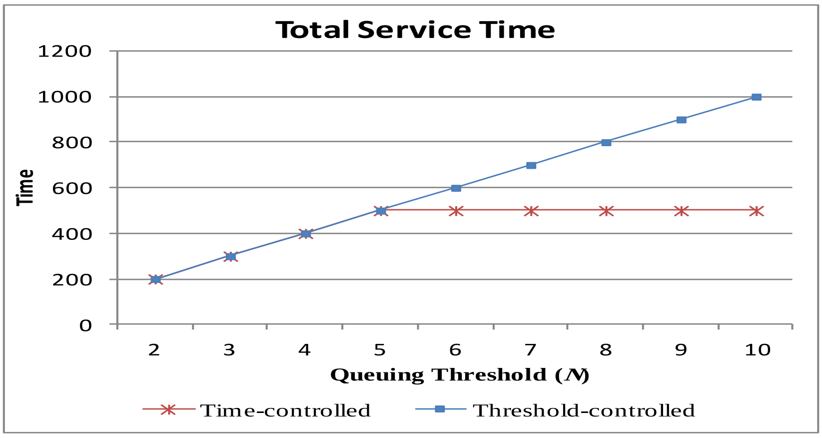

Figure 5: Total service time for different kind of queuing, threshold-controlled and time-controlled.

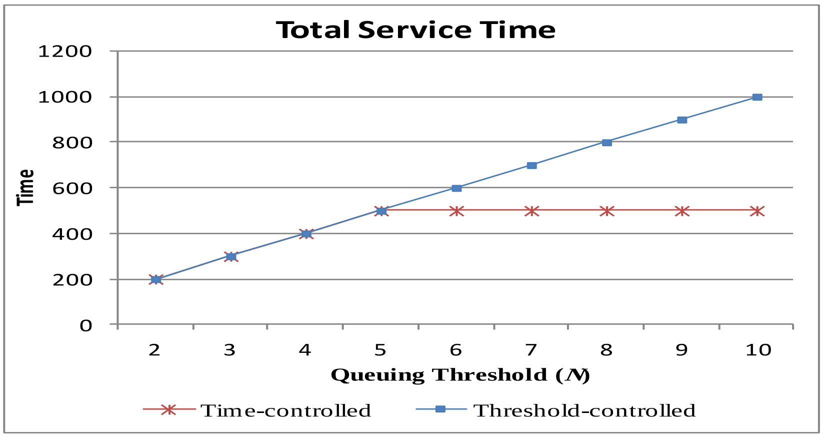

Figure 5 shows the total service time for various thresholds where the queuing threshold per powerful nodes ranges from 2 to 10 messages. The values are determined analytically using Equation (4.13). When the queuing threshold is used solely to control the transmission process, the total service time is increasing linearly with the increase of the queuing threshold. On the other hand, if time is used to control the transmissions in case of larger queuing threshold the result shows a great improvement of up to 50% in the tested scenarios. At the start when the queuing threshold is relatively small, time controlled environments show no improvements in performance while imposing time threshold will improve the performance by reducing the delay of up to 50%.

Most of the energy is consumed during cycle transition. To improve energy consumption, the number of cycles should be minimized without increasing the end-to-end delay. Therefore, the transitions from Send state to *Listen* state and vice versa within the powerful node is reduced by increasing the queue threshold (N) because the time taken for the buffer to be filled with the threshold number of packets for high values of N is more when compared to low values of N. Here, the end-to-end delay is at risk in case of large queuing threshold especially in case of low arrival rate.

Figure 6 demonstrates the improvement in energy consumption when queuing is used. When queuing is used, the number of cycles are reduced which will result in power conservation. The figure shows that with the increase of queuing threshold the power consumption will decrease. Those scenarios assume the queuing threshold per powerful nodes ranges from 2 to 10 messages. The values are determined analytically using Equation (4.18) and (4.19) for queuing-less and queued environments respectively. When the queuing threshold is used to control the transmission process, the power consumed increases linearly with the increase of the queuing threshold. The improvement in power consumption when time-controlled queuing is used can be clearly seen in Figure 6 when looking at slope-intercept form of both scenarios where the slope is improved by half of its original value.

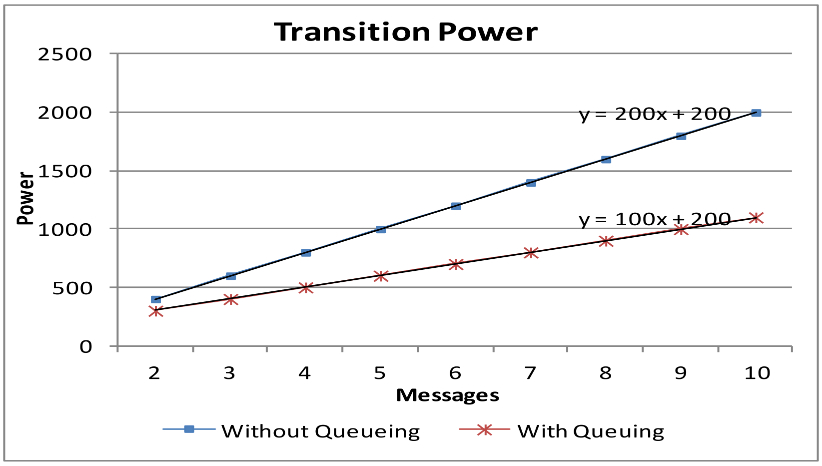

Figure 6: Power consumption versus number of messages equal to the queuing threshold.

6. Conclusion

In this paper, we improved the dual-state approach by dividing the active state to three different modes and postpone the transmission until the queuing threshold has been reached or a specific time has passed whatever comes first. The use of different states allows the sleeping nodes to reduce their power consumption to the lowest possible level will staying connected. To reduce the power consumption during the active states, time-controlled queuing mechanism is utilized to extend the listening mode and avoid increasing the delay in a low arrival rate environment. We have developed an analytical model using queuing theory for a cluster based sensor network by using M/G/1 queuing model and study the system performance in terms of average energy consumption and end-to end delay. Our results demonstrate that the average energy consumption can be reduced by up to 50% depending on the value of the queue threshold. This technique just like all previous ones will increase the average delay of the system which makes it applicable to delay tolerant applications. When we compared time-controlled against threshold-controlled queued HSN, the time-controlled reduced the delay by up to 50% compared to the threshold-controlled environment. As a future extension of this work, studying the impact of end-to-end delay and energy consumption in a controlled environment using a test-bed or a real experiment will reveal important observations.

Refe rences

[1] Akkaya, K. and M. Younis, *"A survey on routing protocols for wireless sensor networks."* Ad Hoc Networks, **3**: p. 325--349, 2005.

[2] Carle, J. and D. Simplot-Ryl, *"Energy-efficient area monitoring for sensor networks."* Computer, **37**(2): p. 40-46, 2004.

[3] Chiasserini, C.F. and M. Garetto, *"An Analytical Model for Wireless Sensor Networks with Sleeping Nodes."* IEEE Transactions on Mobile Computing,, **5**(12): p. 1706-1718, 2006.

[4] Zhi, Q., A. Subramanian, and A.H. Sayed, *"REACA: An Efficient Protocol Architecture for Large Scale Sensor Networks."* Wireless Communications, IEEE Transactions on, **6**(8): p. 2924-2933, 2007.

[5] Yarvis, M., et al. *"Exploiting heterogeneity in sensor networks*." in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, 2005.

[6] Lewis, G., et al. *"A system for simulation, emulation, and deployment of heterogeneous sensor networks*." in *Proceedings of the 2nd international conference on Embedded networked sensor systems*. Baltimore, MD, USA: ACM, 2004.

[7] Maheswar, R. and R. Jayaparvathy, *"Power Control Algorithm for Wireless Sensor Networks using N-Policy M/M/1 Queueing Model."* International Journal on Computer Science and Engineering, **2**(7): p. 2378-2382, 2010.

[8] Colette, J. and N. Le Huy, *"Robust self-stabilizing weight-based clustering algorithm."* Theor. Comput. Sci., **410**(6-7): p. 581-594, 2009.

[9] Bertsekas, D. and R. Gallager, *Data Networks*. Second ed: Prentice Hall. 592 pages. 1992.

[10] Kleinrock, L., *Queueing Systems, Volume 2: Computer Applications.*, New York: John Wiley & Sons, Inc. 549 Pages. 1976.

[11] Ross, S., *Introduction to Probability and Statistics for Engineers and Scientists*. Second ed: Academic Press. 512 pages. 2000.

[12] Jiazhen, Z. and M. Kenneth. *"A delay based multipath optimal route analysis for multi-hop CSMA/CA wireless mesh networks*." in *Proceedings of the 20th international teletraffic conference on Managing traffic performance in converged networks*. Ottawa, Canada: Springer-Verlag, 2007.

[13] Joseph, P., H. Jason, and C. David. *"Versatile low power media access for wireless sensor networks*." in *Proceedings of the 2nd international conference on Embedded networked sensor systems*. Baltimore, MD, USA: ACM, 2004.

[14] Heinzelman, W.R., A. Chandrakasan, and H. Balakrishnan. *"Energy-efficient communication protocol for wireless microsensor networks*." in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, 2000.