An Adaptive N-Policy Queueing System Design for Energy Efficient and Delay Sensitive Sensor Networks

Jie Chen
$cencen\_cj2015@yahoo.com$

Biplab Sikdar
National University of Singapore
bsikdar@nus.edu.sg

Mounir Hamdi
Hamad Bin Khalifa University
hamdi@hbku.edu.qa

***Abstract*—This paper considers the problem of energy-delay tradeoff in wireless networks using a N-policy queueing system based scheduler. A novel analytical model for N-policy queueing system is proposed and tested against other established models and simulation results. Using the model, we argue that N-policy queueing system does not necessarily save more energy as N increases.Based on the analytical and simulation results, we present a scheme for the optimal selection of N for a given arrival and service rate. Simulation results based on applying this framework on sensor networks show that the proposed schemes outperforms previous work in the area. Furthermore, an adaptive N-policy system design is illustrated and shown to save energy while satisfying delay requirements.**

I. Introduction

The continued growth in the popularity of information and communication technologies and the resulting impact of their power requirements has led to the birth of the concept of green networks. Due to the relatively short lifetime of the battery that typically powers the end terminals, how to optimize the energy usage to facilitate reliable and energy efficient wireless communication has been a critical research issue in both academia and industry. Much of the existing work in this focuses on sleeping period control for saving energy [4], [5]. However, while increasing the sleeping period increases the energy savings, the average packet delay also increases. This gives rise to another concern in the system design: the energy-delay tradeoff.

This paper presents a scheduler to address the energy-delay tradeoff in wireless devices. This paper proposes an analytical model to characterize the fundamental tradeoffs between energy saving and latency in wireless devices. Henceforth, in our work, the wireless devices together with the aggregation point compose a basic sensor system. The packets are carried from the physical memory of the devices to the transmitter (i.e. scheduling a packet for transmission) by proactively sensing the workload in the device's packet queue.

Sensor systems have their applications in many areas. It is widely used in home surveillance, supermarket self-checkouts and other civil applications. The limited battery power of sensors has been a constraint in their design. Together with the expected customer experience of a sensor system, their power management scheme has become a major metric to measure the system's performance. Power management schemes vary in different scenarios. For inter-device communication in the system, the transmission power is adjusted against the condition of the communication channel. Also, the power of the devices may be switched between on and off modes according to specific design configurations. Much work has been conducted for the latter case [4], [5], [7]–[9]. To determine how to switch between different power modes, the criteria may be the duration of the working hours [4], [5], the amount of workload that has been processed, or the amount of workload to be processed [7], [8]. In another line of work, N-policy queue to schedule on-off periods has been considered. The N-policy queue has been studied extensively in the past [2], [3]. How to find the optimal N to achieve a system design goal while meeting the required statistical performance indicators related to customer experience has been a practical engineering issue. Existing works typically evaluate the long term queue length distribution and use Little's law to obtain the average customer waiting duration in the queue, and furthermore the average drop ratio. To be more specific, the long term expected queue length is decomposed into two parts : the working part when the server is on and the idle part when the server is off. However, these works are primarily theoretical and inadequately argue for their applicability in real life scenarios. For example, reference [4] aims to design a power efficiency scheme for wireless sensor networks but the scheme neglects the finiteness of the buffer size of each sensor. Furthermore, most of the existing work assumes that the traffic generated by the nodes is well known in advance and can be modeled accurately. However, traffic in real-life systems can vary over time and are in general unpredictable.

Our work aims to change the system configuration to adapt to the real-time traffic in order to save energy while not violating the delay constraint. In the proposed system, we dynamically select the threshold N in a a N-policy queue. Intuitively, when the traffic is high, the system should resume work more quickly (i.e. the threshold N should be small), and vice versa. It is also our objective to evaluate the long term average duration that the system spends in both sleeping and active modes. Using these average on-off times, we derive the system performance metrics and consequently the threshold N, which can be adjusted iteratively.

In short, our work has four contributions:

- A new analytical model is derived and tested against the previous models in literature and also against simulation results.
- Based on the analytical model, power consumption for a selected threshold is derived. Both analytical derivation and simulation results show that the power consumed does not decrease as the threshold N increases, given a fixed arrival and service rate.
- Comparisons of performance against the established duty cycle control scheme are conducted and show positive results.
- A threshold N adjustment scheme is proposed and shown to save more energy within a given delay bound.

The rest of the paper is organized as follows. Section II presents the related work. Section III presents the system model, its evaluation, and its comparison against simulation results. Finally, Section IV concludes the paper.

II. Literature Review

Previous researches focus on the derivation of the system queueing analysis when the system is steady, given a well-established statistical traffic model. Literature has also covered how to model the network traffic. This paper does not consider the issue of traffic modeling.

Research on the duty cycle control for sensor system (T-policy queueing theory) has included the following work: [4], [6]. In [4], the policy manages to achieve limited amount of backlog in the buffer by using the control principle while [6] touches the base of delay issue. It increases the duty cycle for some nodes to decrease the delay, as these nodes are far from the sink and have more energy in storage.

Researches that follow the case of N-policy queueing theory includes: [7], [8]. Different cost functions for the queues have been proposed in literature. In [7], the cost function composes of packet holding energy consumption while in [8], the cost function includes transition power per cycle. It is worthwhile to note that in [7], if only considering the on mode energy consumption and off mode energy consumption, the power per threshold is a constant while in reality, as demonstrated well by simulation and mathematical analysis, the power per threshold varies and does not follow a deterministic pattern. Furthermore, to set the benchmark power as the one while $N = 1$ is not reasonable as $N = 1$ means the system would switch to sleeping mode when the buffer is empty. The correct benchmark would be the value when the system is always on, that is when $N = 0$. Research on N-policy queue theoretically could be categorized as N-policy with setup time and N-policy without setup time. The former has its applicability to the real world situation when the server needs time to warm up before the service while for the latter case, most of the work is devoted to generalize the policy into broader representations, such as to consider the arrival as batch Poisson process or to allow for the service time to follow a general probability distribution [3]. In [2], the author has presented the mathematical analysis of system performance metrics for different N-policy queueing policies. In [3], the authors have discussed further the mathematical models behind the queue length distribution incurred by adding the threshold. Another interesting branch that ascends from the N-policy queue is to optimize the system from the user's perspective, that is to grant the user's strategic decision whether to enque or not based on the current state of the queue [10]. Most of these work on N-policy is for server with finite capacity. Works that discuss the tradeoff between energy conservation and average packet delay are largely absent.

III. System Model

Let the packet arrival process at each device follow a Poisson distribution with average rate λ and the scheduler departs the packets at a steady rate of μ. The maximum queue length at each node is K. The queue status is observed every $\frac{1}{\mu}$ interval and we denote the threshold for waking up the queue by N.

A. Model Jessica

The probability that the queue length reaches the threshold during the first observation interval is given by

$$P_1 = 1 - \sum_{0}^{N-1} \exp(-\frac{\lambda}{\mu}) \cdot \frac{\frac{\lambda}{\mu}^k}{k!}. \quad (1)$$

Similarly, the probability that the queue length reaches the threshold during the second observation interval is given by

$$P_2 = (1 - P_1) \cdot (1 - \sum_{0}^{N-1} \exp(-2 \cdot \frac{\lambda}{\mu}) \cdot \frac{(2 \cdot \frac{\lambda}{\mu})^k}{k!}). \quad (2)$$

Proceeding along these lines, the probability that the queue length reaches the n-th observation interval is given by

$$P_n = \prod_{k=1}^{n-1} (1 - P_k) \cdot (1 - \sum_{0}^{N-1} \exp(-n \cdot \frac{\lambda}{\mu}) \cdot \frac{(n \cdot \frac{\lambda}{\mu})^k}{k!}). \quad (3)$$

The average number of observation intervals for the queue length to reach the threshold N is as follows:

$$E(s_{idle}) = \sum_{i=1}^{\infty} i \cdot P_i. \quad (4)$$

The average queue length when the scheduler starts the process of transmitting a packet is:

$$\begin{aligned} E(Q_{len}) &= \sum_{i=N}^{K} i \cdot P(Q_{len} = i) \\ &= \sum_{i=N}^{K} i \cdot \sum_{j=1}^{\infty} P(Q_{len} = i, s = j). \quad (5) \end{aligned}$$

where

$$
\begin{aligned}
&P(Q_{len} = i, s = j) \\
&= \sum_{k=0}^{N-1} P(k, \Delta t = (j-1)\cdot\frac{1}{\mu}) \cdot P((i-k), \Delta t = \frac{1}{\mu}) \\
&= \sum_{k=0}^{N-1} \exp -(j-1)\cdot\frac{\lambda}{\mu}\cdot\frac{[(j-1)\cdot\frac{\lambda}{\mu}]^k}{k!}\cdot\exp -\frac{\lambda}{\mu}\cdot\frac{\frac{\lambda}{\mu}^{(i-k)}}{(i-k)!} \\
&= \sum_{k=0}^{N-1} \frac{\exp(-j\cdot\frac{\lambda}{\mu})\cdot\frac{\lambda}{\mu}^i\cdot(j-1)^k}{k!\cdot(i-k)!}.
\end{aligned} \tag{6}
$$

For the active period, let the transition matrix be denoted by P. The dimension of P is $K \times K$ and its elements are given by

$$
p_{ij} = \begin{cases} 1, & \text{if} i = j = 0 \\ 0, & \text{if}(i = 0 \wedge j \neq 0) \vee (j < i-1) \\ \exp(-\frac{\lambda}{\mu})\frac{\frac{\lambda}{\mu}^{j-i+1}}{(j-i+1)!}, & \text{if } (i-1) \leq j < K \\ 1 - \sum_{k=0}^{K-1} p_{ik}, & \text{otherwise} \end{cases} \tag{7}
$$

The initial distribution of queue length is denoted P_{init}, the dimension of which is $1 \times K$. P_{init} is given by

$$
P_{init}(i) = \sum_{j=1}^{\infty} P(Q_{len} = i, s = j). \tag{8}
$$

The probability that the scheduler stops because the queue becomes empty during the first observation interval is given by

$$
P_1(0) = P_{init}(0). \tag{9}
$$

Similarly,

$$
P_i = P'_{(i-1)} \cdot P \tag{10}
$$

where

$$
P'_{(i-1)}(k) = \begin{cases} 0, & \text{if } k = 0 \\ P_{(i-1)}(k), & \text{otherwise.} \end{cases} \tag{11}
$$

The average number of steps for the active period within one cycle is given by

$$
E(s_{active}) = \sum_{i=0}^{\infty} i \cdot P_i(0). \tag{12}
$$

Let $E(Arrival_N)$ be the average amount of packets that arrive during one cycle. Then,

$$
E(Arrival_N) = E(Q_{len}) + \frac{\lambda}{\mu} \cdot E(s_{active}). \tag{13}
$$

Let $E(Depatl_N)$ be the average number of packets that depart during one cycle. Then

$$
E(Depart_N) = E(s_{active}). \tag{14}
$$

Let $E(drop)$ be the expectation of the drop ratio for a running cycle of queue length from 0 to 0. Then,

$$
E(drop) = 1 - \frac{E(s_{active})}{E(Q_{len}) + \frac{\lambda}{\mu}\cdot E(s_{active})}. \tag{15}
$$

Finally, let $E(latency)$ be the expectation of the latency for a running cycle of queue length from 0 to 0. Then,

$$
E(latency) = \frac{E(s_{idle})}{\mu} + (\frac{1}{\mu} - \frac{1}{\lambda}) \cdot \frac{E(s_{active}) + 1}{2}. \tag{16}
$$

1) Model Jessica-simple: It is also intuitively simple to reach the derivation that $E(s_{idle}) \approx \mu \cdot \frac{N}{\lambda}$. Suppose t is the active period when the queue length changes from N to 0. Then, we have

$$
t \cdot \lambda + N = t \cdot \mu \tag{17}
$$

which gives

$$
t = \frac{N}{\mu - \lambda}. \tag{18}
$$

Now, since $t\mu = E(s_{active})$, we have

$$
\begin{aligned}
E(latency) &= \frac{N}{\lambda} + (\frac{1}{\mu} - \frac{1}{\lambda}) \cdot \frac{1 + \frac{N}{\mu-\lambda}\cdot\mu}{2} \\
&= \frac{1}{2\cdot\mu} + \frac{N-1}{2\cdot\lambda}.
\end{aligned} \tag{19}
$$

Thus, so far two expressions for average packet latency are attained. From the above revised model, we can then derive the expected energy consumption of the node as

$$
\begin{aligned}
E(Energy) &= \frac{Power_{on}\cdot E(t_{active}) + Power_{off}\cdot E(t_{idle})}{E(t_{active}) + E(t_{idle})} \\
&= \frac{Power_{on}\cdot\frac{N}{\mu-\lambda} + Power_{off}\cdot\frac{N}{\lambda}}{\frac{N}{\mu-\lambda} + \frac{N}{\lambda}} \\
&= Power_{on}\cdot(1 - \frac{\lambda}{\mu}) + Power_{off}\cdot\frac{\lambda}{\mu}.
\end{aligned} \tag{20}
$$

The expression above suggests that the average power consumed is independent of the value of N. Firstly, this is counter-intuitive and contrary to our intention of the design. Secondly, our simulation results show that there are discernible variations in the power consumption with changes in N. The counter-intuitive analytic result is due to the approximations made in the simplified model.

We now develop an more elaborate model for the average power consumption for a given threshold. Let the overall observation timespan be t, the average busy steps per cycle be BC and the average idle steps per cycle beIC. Then, on average there would be m cycles, with

$$
m = \frac{t\cdot\mu}{BC + IC}. \tag{21}
$$

We consider the power consumption of a node to comprise of two components E_1 and E_2, with $E = E_1 + E_2$. E_1 and E_2 are given by

$$
E_1 = k \cdot [Power_{on}\cdot BC + Power_{off}\cdot IC] \tag{22}
$$

and

$$
E_2 = \begin{cases} Power_{off}\cdot\beta\cdot(BC + IC) & \text{if } \beta(BC+IC) \leq IC \\ Power_{off}\cdot IC + & \text{otherwise} \\ [\beta(BC+IC) - IC]\, Power_{on} & \end{cases} \tag{23}
$$

Thus we have

$$Power = \begin{cases} \frac{k\cdot(Power_{on}BC+Power_{off}IC)}{(k+\beta)(BC+IC)} + \frac{Power_{off}\cdot\beta\cdot(BC+IC)}{(k+\beta)\cdot(BC+IC)} & \text{if } \beta\le\alpha \\ \frac{k}{k+\beta}\frac{Power_{on}BC+Power_{off}IC}{BC+IC} + \frac{[\beta\cdot(BC+IC)-IC]\cdot Power_{on}}{(k+\beta)\cdot(BC+IC)} + \frac{Power_{off}\cdot IC}{(k+\beta)\cdot(BC+IC)} & \text{otherwise} \end{cases} \quad (24)$$

with $\alpha = \frac{IC}{BC+IC}$.

2) Threshold Value and Energy Consumption: We now show that with N-policy, increasing the threshold dose not necessarily lead to energy savings, when considering a fixed time window and variable arrival rates. Consider a scenario with fixed arrival and service rates, when the threshold is increased from N_1 to N_2. It results into two different pairs of (k_i,β_i). We can then have the following possibilities.

Case 1: $\beta_1\le\alpha$ and $\beta_2\le\alpha$. In this case,

$$\begin{aligned} P_{N_1}-P_{N_2} &= \frac{P\cdot k_1+P_{off}\cdot\beta_1}{k_1+\beta_1}-\frac{P\cdot k_2+P_{off}\cdot\beta_2}{k_2+\beta_2} \\ &= \frac{k_1\cdot\beta_2-k_2\cdot\beta_1}{(k_1+\beta_1)\cdot(k_2+\beta_2)}\cdot(P-P_{off}) \end{aligned} \quad (25)$$

Let $N_1=17$ and $N_2=26$. Then, $(k_1,\beta_1)=(5,0.2)$ and $(k_2,\beta_2)=(3,0.5)$, for $t=10$, $\mu=100$ and $\lambda=10$. We then get $P_{N_1}>P_{N_2}$. Now, consider the case where $N_1=17$ and $N_2=29$. Then we have $(k_1,\beta_1)=(5,0.2)$ and $(k_2,\beta_2)=(3,0.1)$, for $t=10$, $\mu=100$ and $\lambda=10$. This results in $P_{N_1}\le P_{N_2}$.

Case 2: $\beta_1>\alpha$ and $\beta_2>\alpha$. In this case,

$$\begin{aligned} P_{N_1}-P_{N_2} &= \frac{(P_{on}-P)\cdot(\beta_1\cdot k_2-\beta_2\cdot k_1)}{(k_1+\beta_1)\cdot(k_2+\beta_2)}+ \\ &\quad \frac{\alpha(P_{off}-P_{on})\cdot(k_2+\beta_2-k_1-\beta_1)}{(k_1+\beta_1)\cdot(k_2+\beta_2)} \\ &= \frac{\alpha\cdot(P_{on}-P_{off})\cdot(\beta_1\cdot k_2-\beta_2\cdot k_1)}{(k_1+\beta_1)\cdot(k_2+\beta_2)}+ \\ &\quad \frac{\alpha\cdot(P_{on}-P_{off})\cdot(-k_2-\beta_2+k_1+\beta_1)}{(k_1+\beta_1)\cdot(k_2+\beta_2)} \end{aligned} \quad (26)$$

Using the same values as N_1 and N_2 as in Case 1, we again reach the conclusion that $P_{N_1}>P_{N_2}$ or $P_{N_1}\le P_{N_2}$.

Case 3: $\beta_1\le\alpha$ and $\beta_2>\alpha$. In this case,

$$\begin{aligned} P_{N_1}-P_{N_2} &= \frac{P\cdot k_1+P_{on}\cdot\beta_1}{k_1+\beta_1}-\frac{P\cdot k_2+P_{on}\cdot\beta_2}{k_2+\beta_2} \\ &\quad -\frac{\alpha\cdot(P_{off}-P_{on})}{k_2+\beta_2} \\ &= (P_{on}-P_{off})\frac{-k_1\beta_2\alpha-\beta_1k_2(1-\alpha)}{(k_1+\beta_1)(k_2+\beta_2)} \\ &\quad +(P_{on}-P_{off})\frac{[\alpha(k_1+\beta_1)-\beta_1\beta_2]}{(k_1+\beta_1)(k_2+\beta_2)} \end{aligned} \quad (27)$$

If $\alpha=0.6$,$(k_1,\beta_1)=(5,0.1)$ and $(k_2,\beta_2)=(1,0.7)$, then we have $P_{N_1}<P_{N_2}$. However, with $(k_1,\beta_1)=(8,0.1)$ and $(k_2,\beta_2)=(1,0.7)$, we have $P_{N_1}>P_{N_2}$.

Case 4: $\beta_1>\alpha$ and $\beta_2\le\alpha$. This same scenario as before can be used for this case.

B. Model Takagi

A conventional method to model the above N-policy problem is to observe the system at time instant when either the server ends its idle period or the server completes a service. The average queue length at an arbitrary time instant is obtained to calculate the average delay and also the drop rate. This model uses mathematical notations from [2]. Details of the model are as follows.

Let ξ_n be the n-th observation instance and

$$\xi_n = \begin{cases} 0, & \text{a idle period ends} \\ 1, & \text{a service completes} \end{cases}. \quad (28)$$

Let L_n be the queue length at n-th observation instance. Then

$$q_k = \lim_{n\to\infty} Prob[\xi_n=0, L_n=k],\ \ 0\le k\le K \quad (29)$$

$$\pi_k = \lim_{n\to\infty} Prob[\xi_n=1, L_n=k],\ \ 0\le k\le K \quad (30)$$

$$q_k = \pi_0\cdot P_{init}(k),\ \ 0\le k\le K \quad (31)$$

$$\pi_k = \sum_{i=1}^{k+1}(\pi_i+q_i)\cdot a_{i,k},\ \ 0\le k\le K \quad (32)$$

$$a_{i,k} = \begin{cases} \exp(\frac{-\lambda}{\mu})\cdot\frac{\frac{\lambda}{\mu}^{k+1-i}}{(k+1-i)!}, & (i\le k-1) \text{ and } (k\ne K) \\ 0, & (i<k-1) \text{ and } (k\ne K) \\ 1-\sum_{n=0}^{K-1} a_{i,k} & \text{otherwise} \end{cases} \quad (33)$$

Let $P=[\pi_0,\pi_1,\ldots,\pi_K]$, $P'=[q_0,q_1,\ldots,q_K]$ and A be a $K\times K$ matrix with $A_{i,j}=a_{i,k}$, $0\le i,j=k\le K-1$. Then,

$$P' = \pi_0 P_{init} = PXP_{init} \quad (34)$$

$$X_{i,j} = \begin{cases} 1, & \text{if } i=j=0 \\ 0, & \text{otherwise} \end{cases} \quad (35)$$

$$A' = TA \quad (36)$$

$$T_{i,j} = \begin{cases} 0, & \text{if } (i\ne j)\vee(i=j=0) \\ 1, & \text{otherwise} \end{cases}. \quad (37)$$

It can be shown that

$$\begin{aligned} P &= (P+P')A' \\ &= P(TA+XP_{init}TA) \end{aligned} \quad (38)$$

Let $B=TA+XP_{init}TA$. Then

$$P = PB. \quad (39)$$

From [2], we then have

$$E(drop) = 1-\frac{1-\pi_0}{\frac{\pi_0\cdot N\cdot\mu}{\lambda}+(1-\pi_0)} \quad (40)$$

$$\begin{aligned} E(latency) &= \frac{\sum_{k=1}^{K-1}k\cdot\pi_k}{(1-\pi_0)\cdot\lambda}+ \\ &\quad \frac{K}{\lambda}\cdot(\frac{\pi_0\cdot N+(1-\pi_0)\cdot\frac{\lambda}{\mu}}{1-\pi_0}-1)-\frac{1}{\mu} \end{aligned} \quad (41)$$

IV. Performance Evaluation

This section evaluates the performance of the proposed model. We first consider a scenario with $\mu = 1000$, $P_{on} = 24.75$ mW and $P_{off} = 0.015$ mW. In Figures 1-3, we show the delay for different values of N for the two proposed models as well as the Takagi model. In addition, we also compare the result based on [8] which is labeled "Jayaparvathy". We note that the Takagi model performs worse when the input rate is low. The difference among the analytical models is insignificant for higher arrival rates and match closely with the simulations.

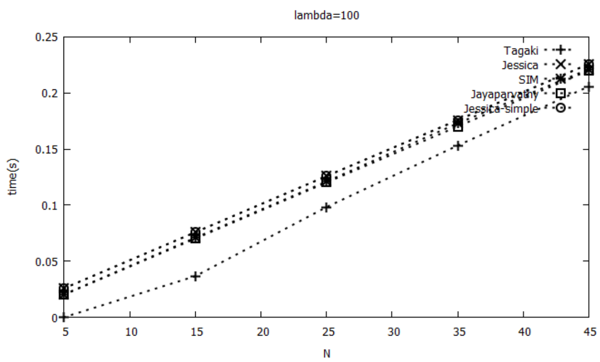

Fig. 1. Average delay when $\lambda = 100$.

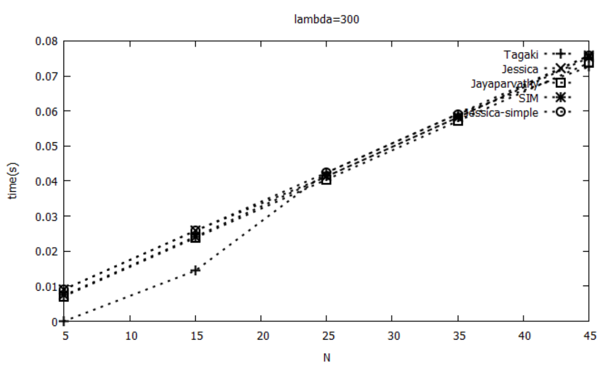

Fig. 2. Average delay when $\lambda = 300$.

Next we compare the performance of the proposed model with that by Byun and Yu [4] using the one-hop network topology from [4]. The comparisons of both delay (Figure 4) and power (Figure 5 shows that the the proposed model outperforms the one proposed in [4].

A. Adaptive Optimal Threshold Selection

Algorithm 1 shows our algorithm for obtaining the optimal threshold for a given bound on the delay. Table I to IV show the optimal threshold values obtained from our algorithm with those from simulations. We observe that the analytic and simulation results have a close match.

Next we consider the case where the traffic arrival rate varies dynamically over time. When the traffic arrival rate

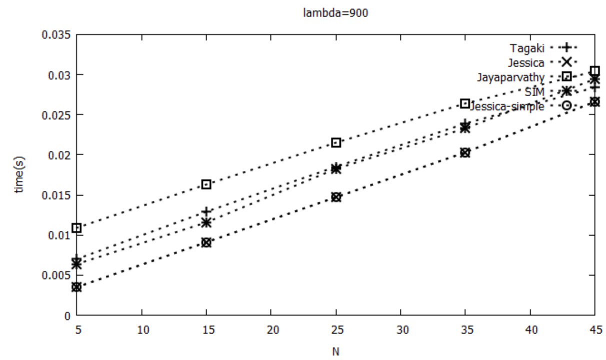

Fig. 3. Average delay when $\lambda = 600$.

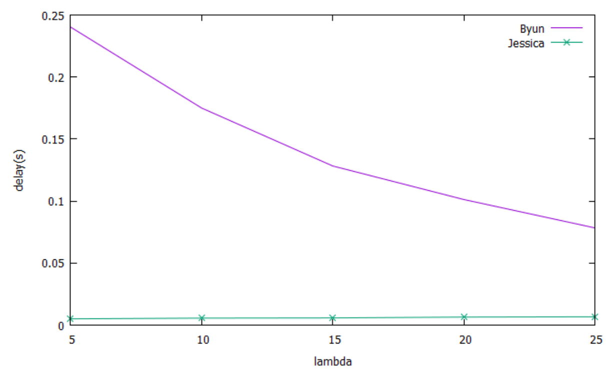

Fig. 4. Cycle Control vs N-policy: Delay

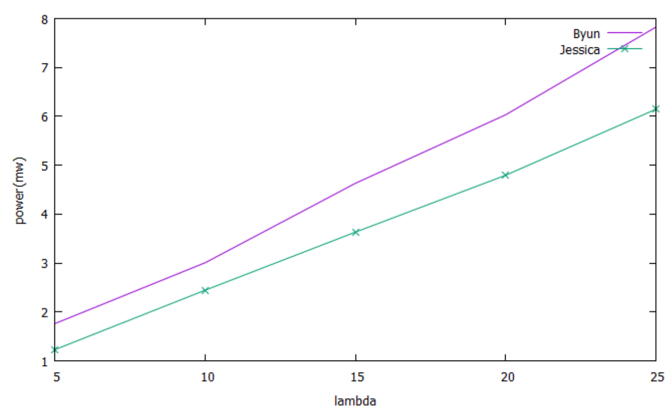

Fig. 5. Cycle Control vs N-policy: Power

Algorithm 1: Optimal N-search Algorithm

Input: D:Expected Latency
Output: N: Optimal N

$temp_p \leftarrow P_{on}$,$opt_n \leftarrow 0$,$t \leftarrow T$
$N \leftarrow \lfloor (D - \frac{1}{2 \cdot \mu}) \cdot 2 \cdot \lambda + 1 \rfloor$ **foreach** n *in* N **do**
  if $Power_n \leq temp_p$ **then**
    $temp_p = Power_n$
    $opt_n = n$
return opt_n

TABLE I
OPTIMAL THRESHOLD: $\lambda = 10$

D	0.5	1	1.5	2
ANA	8	16	27	32
SIM	7	17	25	39

TABLE II
OPTIMAL THRESHOLD: $\lambda = 50$

D	0.1	0.2	0.3	0.4
ANA	9	18	27	36
SIM	7	17	29	37

TABLE III
OPTIMAL THRESHOLD: $\lambda = 100$

D	0.1	0.2	0.05	0.01
ANA	19	38	7	1
SIM	17	38	8	2

TABLE IV
OPTIMAL THRESHOLD: $\lambda = 600$

D	0.01	0.02	0.03	0.035
ANA	7	23	33	42
SIM	11	18	35	38

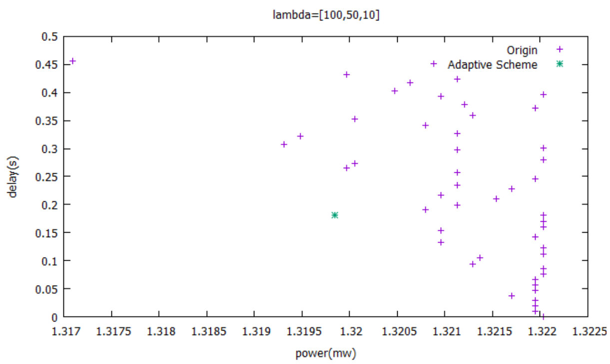

Fig. 6. Delay Upper Bound=0.2

is high, it is expected that the server will wake from the sleeping period sooner. Intuitively, the threshold N would become smaller. The packet scheduler should thus change the threshold value based on the packet input rate. The proposed scheduler periodically monitors the arrival rate. Once there is a change, the scheduler determines the optimal N based on the current traffic information (straightforward metrics are the arrival rate and flow timespan). We now compare the power consumptions for the adaptive thresholding scheme and the fixed (i.e. "original") thresholding scheme, for a given delay bound. Results are shown in Figure 6 and Figure 7. It is noticeable that for the given delay upper bound (0.2 sec), the adaptive scheme performs better than the original scheme.

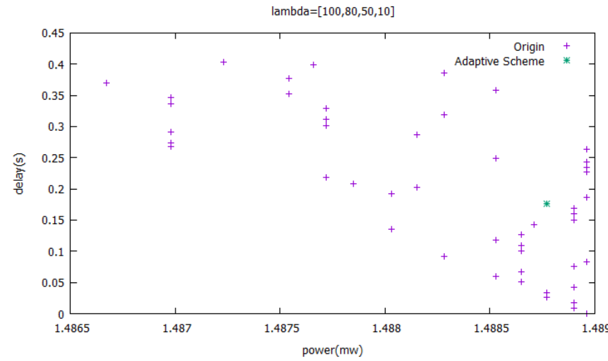

Fig. 7. Delay Upper Bound=0.2

V. CONCLUSION

This paper presented an adaptive thresholding scheme for sensor systems targeting a given upper bound on the delay while minimizing the power consumption. This mechanism is on our proposed analytical model. Simulation results have been used to validate the model. Based on the analytical model, an optimal threshold selection scheme has been devised and shown to give results that match very closely with the simulation results. The threshold selection scheme may be used to adaptively change the threshold when the traffic arrival rate varies. Simulation results show that changing the threshold adaptively with the arrival rate can save energy compared with a scheme that uses a fixed threshold.

REFERENCES

[1] J.R.Norris, "Markov Chains (Cambridge Series in Statistical and Probabilistic Mathematics)" *Cambridge University Press*1997

[2] H.Takagi, "M/G/1/K queues with N-policy and setup times"*Queueing Systems* 1993

[3] H.Lee, B.Ahn and N.Park,"Decompositions of the Queue Length Distributions in the MAP/G/1 Queue under Multiple and Single Vacations with N-policy"*Stochastic Models vol.17 iss.2*2001

[4] H. Byun and J.Yu, "Adaptive Duty Cycle Control with Queue Management in Wireless Sensor Networks"*IEEE TRANSACTIONS ON MOBILE COMPUTING vol.12, no.6* June 2013

[5] Y.Li,K.Chai,Y.Chen and J.Loo, "Duty cycle control with joint optimization of delay and energy efficiency for capillary machine to machine networks for 5G communication system" *Transactions on Emerging Telecommunications Technologies*2015

[6] Z.Chen,A.Liu,Z.Li,Y.Choi and J.Li, "Distributed duty cycle control for delay improvement in wireless sensor networks"*Peer-to-Peer Networking and Applications vol.10, iss.3* May 2017

[7] F.Jiang, C.Yang and F.Lei, "Lifetime elongation for wireless sensor network using queue-based approaches" *The Journal of SuperComputing* March 2012

[8] R.Maheswar 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 vol.2, no.7* 2010

[9] W.M. Kempa, "On Queueing Delay in WSN with Energy Saving Mechanism Based on Queued Wake Uo"*IWSSIP 2014, 21st International Conference on Systems, Signals and Image Processing* May 2014

[10] J.Wang,X.Zhang and P.Huang, "Strategic behavior and social optimization in a constant retrial queue with the N-policy"*European Journal of Operation Research vol.256, iss.3* Feb. 2017