

Performance Evaluation of A Power Management Scheme for Disruption Tolerant Network

Yong Xi, M. Chuah
CSE Department
Lehigh University
Bethlehem, PA 18015

K. Chang
Telcordia Technologies
1 Telcordia Drive
Piscataway, NJ 08854

Abstract

Disruption Tolerant Network (DTN) is characterized by frequent partitions and intermittent connectivity. Power management issue in such networks is challenging. Existing power management schemes for wireless networks cannot be directly applied to DTNs because they assume the networks are well-connected. Since the network connectivity opportunities are rare, any power management scheme deployed in DTNs should not worsen the existing network connectivity. In this paper, we design a power management scheme called context-aware power management scheme (CAPM) for DTNs. Our CAPM scheme has an adaptive on period feature that allows it to achieve high delivery ratio and low delivery latency when used with Prophet, a recently proposed DTN routing scheme. Via simulations, we evaluate the performance of the CAPM scheme when used with the Prophet routing scheme in different scenarios e.g. different traffic load, node speeds and sleep patterns. Our evaluation results indicate that the CAPM scheme is very promising in providing energy saving (as high as 80%) without degrading much the data delivery performance.

Categories and Subject Descriptors:

C.2.2 [Network Protocols], C.4 [Performance of Systems]

General Terms:

Algorithms, Design, Performance

Keywords: power management; disruption tolerant networks; sleep pattern; optimization

1 INTRODUCTION

A Disruption Tolerant Network (DTN) [1] is a network of nodes with frequent partition and intermittent connectivity. Recent research interests in this area include network architecture design and different routing algorithms for DTNs [2][3][4][5]. The routing schemes provide intra and inter-region data delivery services.

Another research effort is related to efficient power

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Qshine'07, August 14-17, 2007, Vancouver, Canada
Copyright 2007 ACM xxx...\$5.00.

management in DTNs. Most wireless devices can operate in different power consumption modes: idle listening, sleeping, transmitting, and receiving. Studies show that idle listening takes high energy consumption, which hence should be reduced and avoided when it is not necessary. The objective of any power management scheme is to minimize nodes' idle listening durations while maintaining network connectivity so that data delivery performance does not degrade.

Power management design in DTNs is very challenging. For example, nodes in sparse network suffer more network partitions. Thus, any deployed power management scheme should not worsen the existing connectivity opportunities. In this paper, we propose an asynchronous power management scheme called the context-aware power management (CAPM) scheme for DTNs. We first describe how the CAPM scheme works. Via simulations, we study how the data delivery performance changes when the CAPM scheme is used with the PROPHET scheme [5]. We provide guidelines on how to select appropriate parameters for the CAPM scheme such that energy savings, delivery ratio and delivery latency requirements can be met. Then, we present two simulation scenarios to illustrate the usefulness of our CAPM scheme and our suggested guidelines in selecting the sleeping parameters. The remainder of this paper is organized as follows. We provide a brief review of related work in Section 2. In Section 3 we present our power management scheme in detail. In Section 4, we describe our simulation setup and present our simulation results. We conclude in Section 5 with discussions on future work.

2 RELATED WORK

2.1 Wakeup scheduling in multi-hop wireless networks

Many wakeup scheduling schemes were proposed for efficient power management in multi-hop wireless networks. In [8], they were summarized into three categories: scheduled rendezvous, asynchronous, and on-demand. In scheduled rendezvous mechanisms [14][15], nodes operate in predetermined wakeup patterns. Clock synchronization is generally assumed in these mechanisms. Asynchronous mechanisms [9][13] do not require time synchronization, but the sleep pattern must be carefully designed to ensure that there are necessary wakeup overlaps for communications. In on-demand wake-up schemes [16][17], often a two-radio architecture is deployed where a second low power radio is used to detect the presence of nodes, and this low power radio will activate a high power radio on demand for packet delivery. However, on-demand schemes are not suitable for

DTNs since the lower power radio often does not have enough transmission and reception ranges to allow nodes that are sparsely distributed to discover one another.

2.2 Routing in Intermittently Connected Networks

Several routing schemes have been proposed for DTNs [2],[3],[5]. Here, we give an overview of Prophet [5], the DTN routing protocol that we used in this paper.

PROPHET. In [5], Lindgren etc designed a probabilistic routing protocol called PROPHET that uses history of encounters and transitivity information for intermittently connected networks. This probabilistic routing scheme establishes a probabilistic metric called delivery predictability at every node a for each known destination b . When two nodes meet, they exchange the delivery predictability information they stored. This information is used to update the estimated delivery predictability to the destination. A message is transferred to the other node if the delivery predictability of the destination of the message is higher at the other node.

2.3 Power Management In DTN

[6][7] present what we believe the first papers that discuss power management issues in DTN. Their papers show that with knowledge on the statistics of the contact like mean and variance, one can significantly improve the performance of the network, like message delivery ratio, delay and energy consumption. However, in many real life scenarios, such statistics may not be readily available, and designing a power management that work in such scenarios is the focus of our work.

3 OVERVIEW OF THE CAPM SCHEME

In this section, we describe our context-aware power management (CAPM) scheme for Disruption Tolerant Networks, which does not require any a priori knowledge on statistics of contact, and handles both neighbor discovery and data delivery.

3.1 Preliminary

We assume that each node independently operates its own wakeup schedule. The sleeping pattern of the CAPM scheme is shown in Figure 1. Each node runs a fixed length duty cycle. In each duty cycle, the node wakes up for a fixed period, W , and then sleeps for the remaining time. The total wake-up cycle period is denoted as C . At the K th wake-up cycle, the node remains on for the full cycle. Thus, a full wake-up cycle consists of K wake-up cycles. We refer to the tuple (W, C, K) as the sleep pattern in this paper.

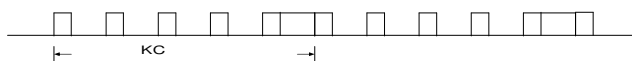


Fig. 1. (W, C, K) sleep pattern

3.2 Neighbor Discovery

Neighbor discovery is used to find neighbors that can be used as potential next-hop nodes for data forwarding. The neighboring nodes that are discovered by a node are referred to as its contacts. To discover neighbors, we assume each node periodically broadcasts a beacon when it wakes up. The beacon message provides information about its node identifier, the time it will remain active. A node that wakes up and have data for delivery piggybacks a delivery notification extension to its beacon message. The delivery notification contains the information about the node identifiers of the destinations of stored messages. An active node that receives a delivery notification with its identifier included will send a delivery accept message to the node that sends the delivery notification.

Via the examples described below, we illustrate how our neighbor discovery scheme works.

In Figure 2-a, all nodes have no data for delivery, so each of them broadcasts a regular beacon message upon waking up. In Figure 2-b, node 1 has data for delivery. When it wakes up, it broadcasts a beacon with a piggybacked delivery notification extension. Since node 2 and node 3 are asleep, there is no reply to node 1's beacon message. But when node 2 wakes up and broadcasts its regular beacon, node 1 will receive this beacon. Then, node 1 can deliver its stored data to node 2. Similar actions are taken when node 3 wakes up later. In Figure 2-c, node 2 has data for delivery. Thus, it issues a regular beacon with piggybacked delivery notification extension upon waking up. Let us assume that node 2 has data for node 1. Since node 1 is active, it will send a delivery acceptance message to node 2 when it hears node 2's delivery notification. Thus, node 2 can discover node 1 and deliver messages to node 1. In Figure 2-d, node 3 has stored data for delivery to nodes 1 & 2. Again, after node 3 sends a regular beacon with piggybacked delivery notification, nodes 1 & 2 can each reply with a delivery acceptance message. Thus, node 3 can discover both nodes and deliver stored messages to them.

In Figure 3, we illustrate how each node maintains a neighbor list after receiving the beacon messages from its neighbors. Let us assume that each node has a fixed on period (W). Upon receiving a beacon from Node 2, Node 1 computes its remaining on time as $(t_1 + W - t_4)$ and sends a delivery accept message to Node 2. Node 2 will insert node 1's information into its neighbor list upon receiving this delivery accept message. Similarly, when node 2 receives node 3's regular beacon at time t_8 , node 2 again updates its neighbor list as shown in Figure 3. We also show in Figure 3 how node 1 updates its neighbor list upon receiving a beacon with piggybacked delivery notification from node 2 and a regular beacon message from node 3.

3.3 Optimal Choice of (W, C, K)

One needs to carefully choose (W, C, K) values such that the delivery ratio, the average delay with and without turning on the sleeping algorithm do not differ by more than say 5%. In addition, we need to make sure that the values are chosen such that we can have the most energy saving. Thus, in this

subsection, we discuss how to derive optimal values of (W, C, K) . Table 2 denotes all the notations that we use. First, W has to be larger than the time it takes for the RF circuit to turn on and transmit a short beacon message to inform active nodes that are within its transmission range. We assume that the minimum value of W , W_{\min} , is about 8-10 ms [11]. C also has to be larger than the time it takes for neighboring nodes to issue beacon messages. So, we assume that $C > C_{\min} = \bar{n}_{eff} * W_{\min}$. For an example, in a network with 40 nodes over $1000 \times 1000 \text{ m}^2$, \bar{n} is approximately 7.3 So, $C_{\min} \sim 73 \text{ ms}$. Next, we derive a relationship between W , C , and K that allows all the packets queued during a full cycle (KC) to be transmitted.

We focus on a particular node, (say N1), and investigate how this node discovers other nodes. We observe that its potential neighbors can be grouped into the following two groups, depending how they are discovered by N1:

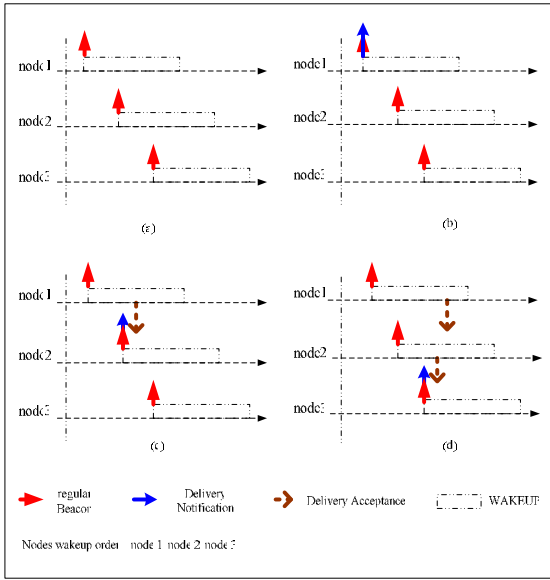


Fig. 2. Neighbor Discovery

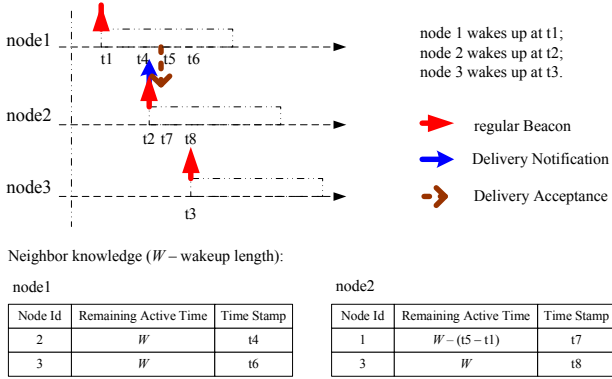


Fig. 3. Construction of Neighbor List

Table 1: Notations Used

W/S	Wakeup /Sleep duration
C	Duty Cycle ($C = W + S$)
r	Sleeping ratio ($r = C / W$)
K	The number of wake-up cycles that constitute a “full wake-up cycle” of a node so that it can discover its neighbors.
R	Transmission range
σ	Node density
\bar{n}	Active number of neighbors per node $\bar{n} = \sigma * \pi * R^2$
L	Packet size
t_{access}	Access delay per packet
θ	Bandwidth
θ'	Effective bandwidth $\theta' = \frac{L/\theta}{L/\theta + t_{access}} \theta$
$f_{i,j}$	Traffic load on link (i, j)
α	Energy consumption unit (e/second), when nodes are in wakeup
β	Energy consumption unit (e/second), when nodes are in sleep
Lm	Number of hops in a route

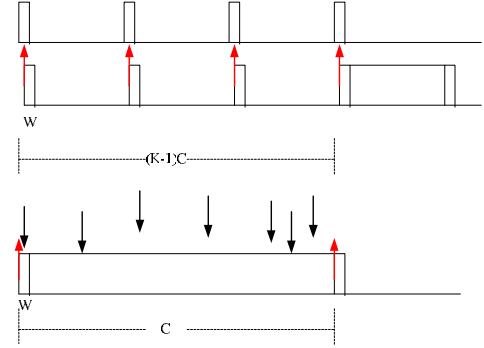


Figure 4: Two cases for discovering contacts

1. First, nodes can be discovered if their awake period is overlapped with the awake period of N1, namely if the timing offset (i.e. awake period offset) between two nodes is less than $2W$, as shown in Figure 4. Suppose the underlying mobility model is random way point, and the contact occurrence is uniformly distributed. Therefore, the average number of nodes that can be discovered within the

$$2W \text{ window is } \frac{2W}{C} \bar{n}.$$

2. Otherwise, nodes can only be discovered during the “full awake cycle”, the K^{th} cycle. The number of nodes that can be discovered in this manner is

$$\frac{C - 2W}{C} \bar{n}$$

During the “full awake cycle”, traffic that contributed by both of the groups needs to be completely sent out. Therefore, we have:

$$(C - C_{\min}) * \theta' > \frac{2W}{C} * \bar{n} * C * f_{i,j} + \frac{C - 2W}{C} * \bar{n} * KC * f_{i,j} \quad (1)$$

At the right hand side of equation 1, the traffic loading due to group 1 and 2 are listed in the first term and second term, respectively. We note that, for group 1, since the wake-up periods are overlapped with that of N1, all the traffic loading in the previous (K-1) cycles has been sent out during the previous (K-1) cycles. Whereas for the nodes that only can be discovered in the last “full wake up cycle”, the traffic loading from the entire K cycles needs to be sent out in the last full wake up cycle..

At the left hand side of equation 1, the amount of traffic that can be supported by the radio is $(C - C_{\min}) * \theta'$, since out of the last wake up cycle of duration C, C_{\min} needs to be used for exchange beacons and can not be used for traffic transmission. θ' is the effected transmission bandwidth when transmitting a packet. We note that when sending a packet over the air interface, the total delay is $t_{\text{access}} + L/\theta$, where L/θ is the packet transmission delay, and t_{access} is the access delay, which includes transmission delay and queuing delay. Therefore, the “effective bandwidth” experienced by a packet

$$\text{is } \theta' = \theta \cdot \frac{L/\theta}{L/\theta + t_{\text{access}}}; \text{ namely due to the access delay,}$$

the effective bandwidth is reduced. We note that t_{access} is dependent on the underlying MAC and loading, and can be determined either from simulations or from analysis presented in [19]. Given most of the operating scenarios of interest at medium traffic load, θ' is approximately 0.3 θ . Details of how this is derived can be found in [20].

To determine $f_{i,j}$ we note the following: let B be the total amount of traffic generated by the source nodes. Therefore, the total offered traffic in this network is BL_m . We assume that this total traffic is uniformly distributed over all N nodes. Thus, each node needs to transmit a traffic of BL_m/N . L_m is dependent on the routing scheme used. For the random waypoint mobility model, one can use the expected meeting time approach described in [18] to derive an estimate of L_m : we note that $\frac{Lm * R}{\bar{v}}$ is the expected meeting time, and, according Eq. 6 of [25], it can be written as $\frac{1}{1.75 p_m + 2(1 - p_m)} \frac{A}{2RL} (\bar{T} + \bar{T}_{\text{stop}})$,

$$\text{where: } p_m = \frac{\bar{T}}{\bar{T}_{\text{stop}} + \bar{T}}, \text{ A is the size of the network area}$$

under consideration, R is the transmission range,, \bar{L} is the mean moving distance of the node, given the random waypoint mobility model, \bar{T} is the mean duration when a node is moving, and, \bar{T}_{stop} is the mean duration when a node is stationary.

Going back to equation (1), using the $f_{i,j}$ value obtained from the above, we have

$$\frac{C - C_{\min}}{2W + (C - 2W)K} > \frac{\bar{n} * f_{i,j}}{\theta'} \quad (2)$$

We refer to the left hand side expression as the Sleeping pattern impact factor (SPIF) and the right hand side expression as the Traffic Threshold Line (THH). We plot SPIF versus different W and C values in Figure 5.

Next, we differentiate the SPIF with respect to C, and obtain

$$\frac{\partial_{\text{SPIF}}}{\partial c} = \frac{KC_{\min} - (2K - 2)W}{[KC + (2 - 2K)W]^2} \quad (3)$$

By setting $\frac{\partial_{\text{SPIF}}}{\partial c} = 0$ and solve for W , and letting W_{thresh}

denote the solution, we have $W_{\text{thresh}} = \frac{KC_{\min}}{2K - 2}$. We

note that if $W > W_{\text{thresh}}$, $\frac{\partial_{\text{SPIF}}}{\partial c} < 0$. Namely, $\frac{\partial_{\text{SPIF}}}{\partial c} = 0$ represents a line parallel to the x-axis with slope equal to 0.

If $\frac{\partial_{\text{SPIF}}}{\partial c} < 0$, SPIF is a family of convex curves with respect to C. Thus, given a value of THH, we want to choose K such that we can have $\text{SPIF} > \text{THH}$ for this family of curves, regardless of the value C. Take K=4 in Figure 5 as an example, if $\text{THH} = 0.1$ (10 flows and 1pkt/sec), then by choosing $W > W_{\text{thresh}}$, all the SPIF curves would lie above the $\text{THH} = 0.1$ line, and hence satisfying equation (2) regardless of the C value.

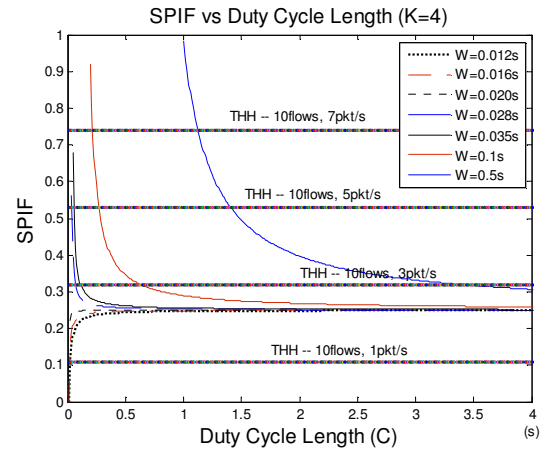


Figure 5: SPIF versus Duty Cycle Length

In addition to loading requirements considered in the last section, a user may wish to realize certain energy saving requirements. Therefore, we determine next the normalized energy consumption relationship to determine further constraints for W, C and K. Let E_e denote the “energy efficient metric”, which is defined as the ratio of the energy consumption under the sleep pattern described in this paper, to the energy consumption without the sleep pattern. Namely, a small E_e represents a high level of energy saving realized by the sleeping pattern. Therefore, E_e can be expressed as:

$$E_e = \frac{\alpha[W(K-1) + C] + \beta(C-W)(K-1)}{\alpha KC}$$

After simplification, we have

$$E_e \propto \frac{1}{r} \left(1 - \frac{\beta}{\alpha}\right) + \frac{\beta}{\alpha} + \frac{1}{K} \left(1 - \frac{\beta}{\alpha} - \frac{1}{r} \left(1 - \frac{\beta}{\alpha}\right)\right) \quad (4)$$

By examining equation (4), we observe that as r and K increases, E_e decreases. In Figure 6, we plot the normalized energy consumption (E_e) with different r and K values. Note that we use $\alpha = 0.179$, and $\beta = 0.0141$ for this plot.

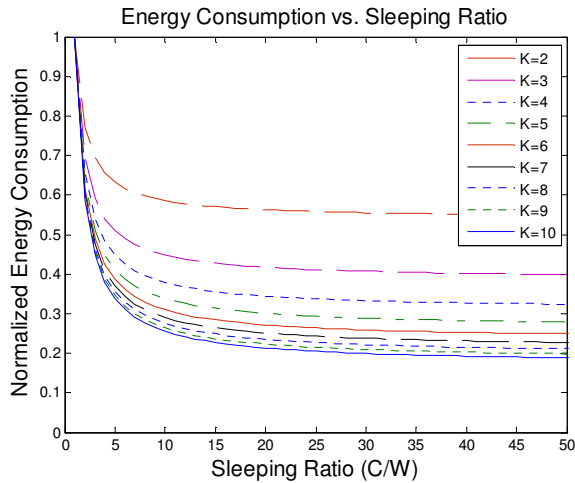


Figure 6: Normalized Energy Consumption vs C/W

In what follows, we provide a set of guidelines to choose the values for (W, C, K) for small THH values which are common for sparse adhoc networks.

Rules for choosing (W, C, K)

1. Choose K and W from Figure 5

Given a traffic load requirement, compute THH. Using Figure 5, choose the largest K such that the convex family of the SPIF curves lies above the THH line. Assuming C/W is large, we have

$$\left(\frac{C - C_{\min imum}}{2W + (C - 2W)K} > \frac{\bar{n}^* f_{i,j}}{\theta'}\right), \text{ when } C/W \rightarrow \infty, \text{ then,}$$

we have $K < \frac{\theta'}{\bar{n}^* f_{i,j}}$. Figure 6 indicates that for energy

efficiency, one would like to have larger K. However, for the average delay consideration, we do not want K to be larger than Kmax (e.g. Kmax=12). Thus, K is chosen to be $\min(K_{\max}, \frac{\theta'}{\bar{n}^* f_{i,j}})$. Once K is chosen, we evaluate

Wthresh and choose W to be Wthresh if this is larger than Wmin.

Example: (1) Given 10-flow, 3pkts/s each, we get $K < 4.8$ using a node density of 1×10^5 nodes/m². Thus, we pick K=4. Then, Wthresh= 10.7 ms (this value does not include about 8-10 ms on/off switching delay [11])

2. Choose C

We use the following criterions to choose C

(i) Energy requirements: Given the chosen value of K and W, we choose a value of r that satisfies the energy requirements

Example: if the energy requirement is $E_e < 0.4$, by consulting Figure 6, given K=4 we observe that r needs to be larger than 10, which makes C~240 ms if W is chosen to be 24 ms.

(2) Delay Constraints: KC has to be less than the one-hop delay requirement, e.g 5s

Example: if one hop delay requirement is 5 seconds, than we have the largest C is 1.25, which is much larger than the required C values from the energy consumption requirement. This means we can choose C between (0.24s, 1.25 second). To achieve better energy efficiency, we would choose C=1.25 second but if we want better delay and delivery ratio, we may choose C=240 ms.

4 PERFORMANCE EVALUATION

4.1 Simulation Setup

In order to evaluate our power management scheme, we implemented our CAPM scheme and Prophet in ns-2 [12]. The performance metrics we used in our evaluation are: (1) *Delivery Ratio*, which is the successfully received number of data divided by the number of total delivered data. (2) *Normalized Energy Consumption*, which is the ratio between the energy consumption with power management and the energy consumption in the absence of power management. This metric is a measure of energy efficiency of our power management scheme, and (3) *Average End-end delay*, which is the delay it takes to deliver a message from the source to the destination.

In our simulation, we use three network scenarios with 40 nodes distributed over (a) 1000x1000m², 2000x2000 m² and (c) 3000x3000 m². The nodes move according to the random waypoint model. We set the pause time to be 10 seconds, and

the maximum node speed to be 5 m/s. All the nodes communicate using a normal transmission range of 250 m and a bandwidth of 1 Mbps. In our paper, we use the same energy parameters in [6] which are replicated in Table 2.

Table 2: Power Usage

State	Idle	Sleep	Transmit	Receive
Power (W)	0.1791	0.0141	0.2818	0.2053

We use constant bit rate traffic as our traffic model. We assume that there are 10 CBR flows. Unless otherwise stated, the sources and destinations of these flows are randomly selected among the 40 nodes. The packet size is 512 bytes and the traffic generation rate for each flow is varied from 0.25 pkts/sec to 3 pkts/sec. We run each simulation for 600 seconds with a warming up period of 1000 seconds and the reported simulation results are based on the average of 5 runs.

4.2 .Optimal Choice of (W,C,K).

In this section, we are interested in demonstrating that the guideline that we provide for choosing optimal (W,C,K) values is useful. In this set of experiments, we use a network with 40 nodes distributed over 1000x1000 m². 10 CBR flows with each flow generating 3 pkts/sec are used. Using the guideline provided in Section III, we select K=3, then we select a W value and determine C_{threshold} from Eqn (2). Next, we simulate three cases with C=0.8C_{threshold}, C_{threshold} and 1.2C_{threshold} for each selected W value. Table 3 tabulates the delivery ratio, the average delay, and the normalized energy consumption for this experiment. It is apparent from Table 3 that when one picks a larger value of C, the average delay value increases but the normalized energy consumption reduces.

W	Delivery Ratio			Avg Delay (sec)		
	0.8C	1.0C	1.2C	0.8C	1.0C	1.2C
0.016	0.9	0.95	0.95	1.82	2.01	2.35
0.018	0.87	0.93	0.94	1.78	1.82	1.86
0.02	0.81	0.93	0.93	1.68	1.81	1.87
0.024	0.82	0.93	0.93	1.61	1.75	1.81

W	Norm. Energy Consumption		
	0.8C	1.0C	1.2C
0.016	0.287	0.279	0.272
0.018	0.313	0.301	0.289
0.02	0.351	0.339	0.321
0.024	0.375	0.347	0.329

Table 3: Results with different Ws

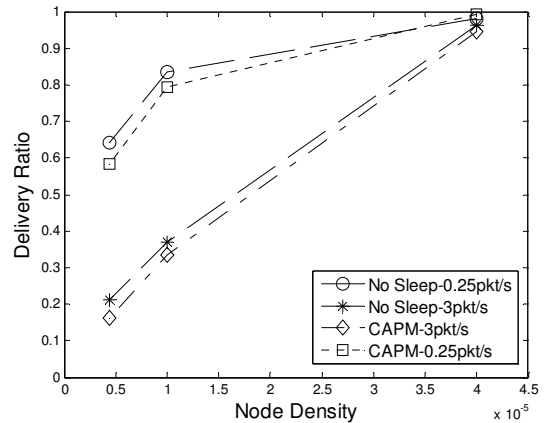
4.3 Impact of Node Densities and Traffic Load

In our second experiment, we investigate the effectiveness of our CAPM scheme when we use different traffic loads and node densities. We tried three network scenarios with different node densities, namely 40 nodes over (i)

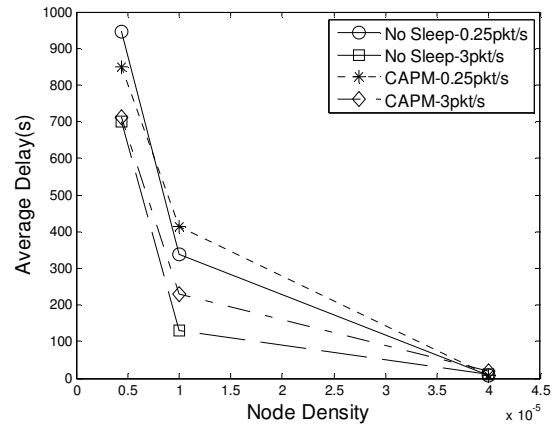
1000x1000 m² (denoted as 1000x1000-40), (ii) 2000x2000 m², and (iii) 3000x3000 m². The parameters used in our simulations are tabulated in Table 4.

Table 4: Parameter values for Experiment 2.

Packet rate	Node density	Sleeping pattern (W, C, K)
3pkt/s	1000x1000-40	(0.024, 1.67, 3)
	2000x2000-40	(0.01, 1.25, 4)
	3000x3000-40	(0.01, 1.6, 3)
0.25pkt/s	1000x1000-40	(0.04, 0.4, 12)
	2000x2000-40	(0.01, 0.4, 12)
	3000x3000-40	(0.01, 0.4, 12)
No Sleep	Beacon period --- 5s	

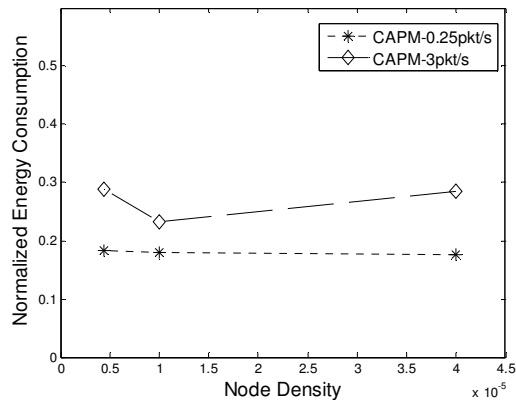


(a) Delivery Ratio versus Node Density



(b) Average Delay versus Node Density

Figures 7(a),(b) &(c) plot the delivery ratio, average delay and normalized energy consumption for our second experiment. From the results, we can see that the delivery ratio does not degrade more than 3% with different node densities when the CAPM scheme is used. However, using CAPM allows us to save 80% energy at low load (0.25 pkts/sec) and about 70-80% energy at high load (3 pkts/sec) with different node densities. We have results using ZebraNet mobility model [20] that show that similar power savings can be obtained.



(c) Normalized Energy Consumption vs Node Density

Figure 7: Impact of Traffic Load and Node Density on CAPM.

Next, we use a network scenario with 40 nodes distributed over 3000x3000 m² and compare the energy efficiency of the CAPM scheme and the PSM scheme described in [7]. We use both a low load (0.25 pkt/s for each flow) and a high load (3 pkts/sec for each flow) scenarios. Our results indicate that CAPM can achieve a 80% energy saving at low load and 70-80% energy saving at high load and yet achieve delivery ratio and average delay that are comparable to the case without sleeping. However, the PSM scheme can only achieve a 40% energy saving. Thus, our CAPM scheme is more effective. Details of such results can be found in our technical report [20].

5 CONCLUSION

In this paper, we present a context-aware power management (CAPM) scheme for DTNs. Our power management scheme allows us to maintain network connectivity while saving energy. Then, via simulations, we study the data delivery performance and energy saving that one can get when CAPM is used together with Prophet. Our results indicate that the CAPM scheme allows us to save energy while maintaining reasonably close data delivery performance to the case without power management. We also investigate how the node densities and traffic load affect the effectiveness of the CAPM scheme. Our results demonstrate that our CAPM scheme is adaptive to different network environments and can provide energy saving of up to 70-80%.

In this work, we only evaluate the impact of power management on the data delivery performance using only one DTN routing scheme. We intend to study the performance of the CAPM scheme with other DTN routing schemes. In addition, we intend to study the impact of different mobility models on the performance of the CAPM scheme with any DTN routing schemes. We also intend to investigate if the same power management scheme can be effective in DTNs with both static and mobile nodes.

Acknowledgements: This work has been supported by DARPA under Contract W15P7T-06-C-P430. Any opinions, findings, and conclusions or recommendations

expressed in this paper are those of the authors and do not necessarily reflect the views of the sponsor of this work.

6 REFERENCES

- [1] K. Fall, "A delay-tolerant network architecture for challenged internets", in Proceedings of ACM Sigcomm, 2003.
- [2] J. Burgess, B. Gallagher, D. Jensen, and B.L.Levine, Maxprop: Routing for vehicle-based disruption-tolerant networks. Proceedings of IEEE Infocom, 2006.
- [3] S.Jain, K.Fall, and R. Patra, "Routing in a Delay Tolerant Network", Proceedings of ACM Sigcomm, 2004.
- [4] M.M.B.Tariq, M. Ammar, and E. Zegura, "Message Ferry Route Design for Sparse Ad hoc Networks with Mobile Nodes", ACM MobiHoc, May22-25, 2006.
- [5] A. Lindgren, A.Doria, and O.Scheln, Probabilistic Routing in Intermittently Connected Networks. In Proc. Workshop on Service Assurance with Partial and Intermittent Resources, Aug, 2004.
- [6] Hyewon Jun, Mostafa H. Ammar, and Ellen W. Zegura, Power Management in Delay Tolerant Networks: A Framework and Knowledge-Based Mechanisms, In Proceedings of IEEE Conference on Sensor and Ad Hoc Communications and Networks (SECON), Sept, 2005.
- [7] Hyewon Jun, Mostafa Ammar, Mark Corner, and Ellen Zegura, Hierarchical Power Management in Disruption Tolerant Networks with Traffic-Aware Optimization, In ACM Sigcomm Workshop on Challenged Networks (CHANTS), Sept, 2006.
- [8] T.Armstrong, "Wake-up Based Power Management in Multi-hop Wireless Networks", Term Survey Paper, University of Toronto.
- [9] R.Zheng, J.C.Hou, and L.Sha, Asynchronous Wakeup for Ad Hoc Networks, MobiHoc'03.
- [10] L.M.Feeney and M.Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc network", Proceedings of IEEE INFOCOM, 2001.
- [11] Power Saving in 802.11: <http://www1.ietf.org/mail-archive/web/manet/current/msg03227.html>
- [12] "The network simulator ns-2", [Online] at <http://www.isi.edu/nsnam/ns/>.
- [13] Yu. Tseng, C. Hsu, and T. Hsieh, Power-Saving Protocols for IEEE 802.11 -Based Multi-Hop Ad Hoc Networks", Proceedings of IEEE Infocom, 2002
- [14] B. Chen et al, "Span: an energy efficient coordination algorithm for topology maintenance in ad hoc wireless networks", Proceedings of 7th ACM, Mobicom, 2001.
- [15] B. Awerbuch, et al, "The pulse protocol: energy efficient infrastructure access", Proceedings of IEEE Infocom, 2004.
- [16] E. Shih, P. Bahl, M. J. Sincalir, "Wake on wireless: An event driven energy saving strategy for battery operated devices", Proceedings of ACM Mobicom, 2002.
- [17] M. J. Miller, N. H. Vaidya, "Power save mechanisms for multihop wireless networks, Proceedings of 1st International Conference on Broadband Networks", 2004.
- [18] T. Spyropoulos et al, "Performance Analysis of Mobility-Assisted Routing", Proceedings of ACM Mobicom, 2006.
- [19] M. Carvalho, J.J. Garcia-Luna-Aceves, "Delay Analysis of IEEE 802.11 in Single-Hop Networks", Proceedings of IEEE ICNP, 2003.
- [20] Y. Xi, M. Chuah, K. Chang, "Performance Evaluation of a Power Management Scheme for Disruption Tolerant Networks", CSE Technical Report, Lehigh University, May, 2007.