

An Energy-Aware Simulation Model and Transaction Protocol for Dynamic Workload Distribution in Mobile Ad Hoc Networks¹

Farhad Ghasemi-Tari, Peng Rong and Massoud Pedram
University of Southern California
{tari, prong, pedram}@usc.edu

Abstract

This paper introduces a network simulation model for detailed evaluation of the performance of different energy management policies in a MANET. Next it presents an energy-aware network transaction protocol that dynamically redistributes the computational workload among a set of cooperative hosts within a MANET so as to improve network performance (network lifetime and service latency.) Extensive simulation data and empirical results are presented and discussed.

1. Introduction

Many of the mobile information processing and communication devices (mobile hosts) in a MANET are battery powered, yet they have to achieve a certain level of performance and remain operational (and network connected) for some desired period of time. Hence, power consumption and clock frequency are important criteria in designing these hosts. Apart from static (compile-time) design optimizations of the hosts, it is possible to improve the performance and lifetime of an ad hoc network of such hosts by employing dynamic (run-time) resource and power management. For example, it should be evident that minimizing the power consumption or maximizing the service speed of any given host may not be sufficient to achieve the lowest latency and longest lifetime for the network, as a whole. The key intuition is that a network of hosts in which the workload and network traffic of every host (thus the power consumption) is uniformly distributed will have the longest network lifetime. Of course, care must be exercised to ensure that the resulting latency for servicing requests meets an upper bound constraint.

Scheduling packet transmission and channel assignment based on dynamic topology changes of the network and residual battery energy capacity of each host and subject to a given quality of service is a key design consideration in the MANET. Efficient utilization of the battery energy capacity has an important influence on the overall performance of the network. Since an energy-depleted host can only be recharged after leaving the network, this event causes an increase in service latency or even service interruption, thus making battery depletion of mobile hosts an undesirable occurrence. Uneven distribution of energy resources and computational workloads is especially harmful from the perspective of prolonging the network lifetime (equivalently stated, delaying the onset of the very first host death). Therefore, it is

favorable to use the residual energies of hosts carefully so the energy-rich host can pick up some of the workload of an energy-thin host [4][15][16]. This is quite important if the workload distribution in the MANET is unbalanced in the first place, and thus, when servicing these tasks in the network, some nodes will have a much higher energy depletion rate than others.

Models developed for power management of MANETs usually consider two cases: selfish users and cooperative users. In the first type of models, users are acting to further their own interest, rather than the system as a whole, while in the cooperative models users are willing to accept external service requests from other hosts in order to maximize the overall network lifetime. The most common approach to evaluating the impact of the users' selfish behavior on the power consumption of MANETs is through the use of economic analytical models [2][3][6][12][18]. In the second type of models the approaches are mostly devoted to the analysis of packet transmission scheduling and channel assignment, where mostly the problems associated with packet routing have been considered [5][9][10][11][14].

The problem of evaluating power management policies in MANETs is often modeled with a network of queues. Analytical results are only possible for some simple Markovian queuing networks and evaluation of many practical Markovian queuing networks does not lend itself to analytical approaches. Indeed the non-Markovian stochastic nature of the dynamic and distributed power management problem, even with a simplified network model, curtails the effectiveness of analytical approaches. The problem that we consider here can be modeled as a non-Markovian queuing network where the only known solution approach is the use of a Monte Carlo random variable generator in a system simulation environment [1][8][13].

In this paper, we present a simulation model of a MANET with dynamically changing topology, where each host has locally and externally generated service requests, and possibly different initial battery energies. The proposed energy-aware network simulation model provides a powerful tool for analysis of models representing real world characteristics of MANETs, and aids in developing optimal policies based on state changes in the network, which can be beneficial for implementation by the network users. We also present an energy-aware network transaction protocol that aims at balancing the computational workloads in order to maximize the network lifetime of the MANET. Its performance is evaluated by our simulation model under a variety of network conditions allowing us to derive the optimal performance conditions.

The rest of this paper is organized as follows: Section 2 provides the related background of the MANET. Section 3 presents the simulation model in details. In section 4, an energy-aware network transaction protocol is developed. Simulation results are presented in section 5. Finally, we conclude the work in section 6.

¹ This research was supported in part by DARPA PAC/C program under contract DAAB07-02-C-P302 and by NSF under grant no. 9988441.

2. Background on MANET

A MANET is an autonomous system consisting of wireless mobile hosts and routers that are not connected by any fixed network infrastructure. The hosts/routers are free to move randomly and organize themselves arbitrarily. Therefore, the network topology changes dynamically and unpredictably over time. Such a network may operate in a standalone mode, or may be connected to the larger Internet. At any given time, the network topology is dynamically determined based on the hosts' geographical positions, their transceiver coverage patterns, transmission power levels, and co-channel interference levels.

The MANET topology changes as mobile hosts move about (mobility) or exhaust their battery source (energy depletion.) Furthermore, due to medium access control (MAC layer protocols), multi-path fading, noise, and signal interference, the capacity of a wireless link may be degraded over time. Ad hoc networks require the support of multi-hop communication schemes in order to relay messages through other hosts when there is no direct access between the sending and receiving hosts.

To support ad hoc mobile communications under the influence of the above-mentioned factors, a communication protocol will need to perform four functions, namely:

- Determining/detecting the dynamic network topology,
- Maintaining network topology/connectivity,
- Scheduling of packet transmission and channel assignment,
- Routing the data packets.

In a MANET, energy conservation for mobile hosts is one of the most important design issues. This is not only because a "dead" host cannot provide service to the local service requests, but also because a "dead" host cannot relay messages from another host to its destinations, hence, adversely affecting the network connectivity and service completions for other hosts in the network. In fact, as the number of available (up and running) hosts decreases, the network may even become disconnected.

The power consumption of a mobile host consists of the communication power and computation power. Communication power refers to the power used by a host to communicate with the other mobile hosts. A mobile host also expends computation power to provide service to various requests and execute network algorithms. Many energy-aware routing protocols have been developed to reduce or balance communication power among mobile hosts [9][10][14].

In this paper, we propose a power management protocol, which is focused on redistributing the spare energy resources to execute computational workloads in a MANET so as to extend the network lifetime. This protocol may be integrated with other network lifetime prolonging techniques in a wireless network protocol stack.

3. Simulation Model

When considering real-world situations in a MANET, a reliable study can only be conducted through the use of a network simulation model. Given that the modeling of most real-world network conditions is a difficult and error-prone task and that the network protocols are quite complex, purely analytical methods have shown only limited benefit. For a complex wireless MANET with interacting hosts, and with stochastically changing network conditions and network topology, NS provides an attractive solution for performance and network lifetime analysis.

In a complex network configuration, the state of art of mathematical statistics puts a damper on the use of analytical approaches due to the lack of determining distribution functions of a random variable defined as a complex convolution form of two or more random variables. For example, in our network model, we must cope with three random variables, two of which must conditionally be convoluting to determine the other random variable. Only for some very specific cases, can one analytically determine the statistical distribution of this random variable. Furthermore, even if we determine this distribution through profiling and approximating polynomials, we cannot use an analytical approach for evaluating network protocols because the state of a network is changing dynamically. In contrast, in the NS environment, we can use Monte Carlo random variable generation and thereby determine the convolution of any number of random variables in any form. Furthermore, for any complex network configuration and complex interactions among network components, discrete event simulation models provide an environment, which is very similar to actual operation of the network in real-world situations, and therefore, by adopting and exercising a network simulation model, we can reliably and easily evaluate the performance of the network.

In this paper, we report an energy-aware network simulation (ENS) model for a wireless MANET, employing the GPSS [7] simulation language, to reveal the power of ENS models for evaluating network protocols under dynamically-varying conditions. More precisely, we have developed a simulation model of a mobile network in which the network topology is changing randomly and the state of the network is changing stochastically throughout the simulation time.

We consider a network of mobile computer systems, which are wireless, connected through a WLAN protocol (e.g., IEEE 802.11b). Each host in the MANET consists of compute and storage resources, communication ability, and a limited energy source in the form of a battery. It is envisioned that the network of hosts is designed to perform a collaborative problem solving task, for example, attempting to contain and suppress a fast moving forest fire, or to provide situational and tactical awareness to the first respondents in a chemical/biological spill situation.

Under the assumption that, hosts are cooperative, we study the effects of redistributing energy on the two network performances:

- The *network lifetime* (also referred to as *end-to-end throughput*) as characterized by the total number of service requests executed throughout the network life time,
- The *latency* in processing service requests as measured by the average waiting time for a request to receive service from some host in the network.

3.1 Components of the Simulation Model

Our simulation model considers a network consisting of multiple mobile hosts, with dynamically changing topology in which each host has some initial battery energy capacity. The system components of each host include a local request queue (LQ) followed by a (low-cost, 8-bit microcontroller) PIC processor (LPIQ), an external request queue (EQ) followed by another PIC processor (EPIQ), and a service queue (SQ) followed by a service provider (SP) (cf. **Figure 1**) Local and external requests arrive at each host according to known statistical distributions. The processing times of PICs are also random variables with known statistical distributions. The output of the LPIQ can either be forwarded to the SQ for local processing or may be sent out for

remote processing by the other mobile host (this process is also called *service distribution*.) Similarly the output of EPIQ may be sent for local processing or may be rejected (dropped) altogether.

External requests that arrive at a mobile host have two distinct types: 1) external requests that are sourced at some other host and 2) external requests, which were initially sent off by the host itself, but were rejected by the remote host and are thus returned back to the host. The second type of external requests is assigned high priority and will be processed by the host locally.

The SP has two states: a low service rate state (low power dissipation and low performance) and a high service rate state (high power dissipation and high performance.) For our modeling purpose, we assumed the following state change policy for the SP. As long as the number of waiting requests in the SQ is lower than some a priori, fixed threshold (say 50% of the queue capacity), the SP stays in the low service rate state. Otherwise it switches to the high service rate state. Once the queue size drops below the threshold, the SP enters low service rate state again. Both service rates are random variables with known statistical distributions. Note that when the maximum capacity of the queue is exceeded, the PIC processors will block the incoming requests, which will be either rejected (if external) or simply dropped (if internal).

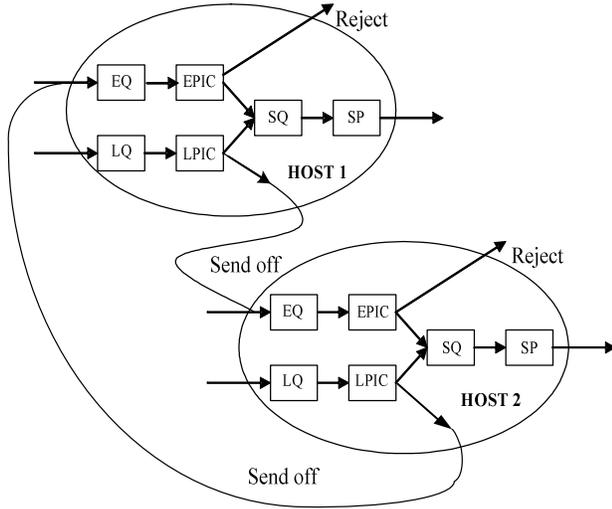


Figure 1. A schematic view of the envisioned MANET.

A host in the MANET must be aware of its surrounding environment and the neighboring hosts with which it can directly communicate. In our simulation model, we consider the transmission energy consumption to increase as a function of square of the distance between two hosts (this is actually quite optimistic, a third or even fourth order dependence may be used instead. [17]) Furthermore, we assumed that the computational energy consumption increases linearly as a function of the required processing time for a service request. The location as well as information about the remaining battery energy of the sending host is transmitted to the receiving host with sending packets for processing. The receiving host can either process the job or send it back depending on its local state and policy under consideration.

3.2 Network Performance Figures of Merit

To study the effect of a policy on the network performance, we adopt two performance metrics as figures of merit: *end-to-end*

throughput and *latency*, which are widely used in wireless networks. We let our simulation model run until a percentage of the mobile hosts expends all of their battery energy capacities. The total number of requests that are serviced in the network provides the throughput of the network whereas the average waiting time of requests in the service queues captures the latency per request. In our simulations, we considered different energy budget constraints for the hosts and experimented with different distributions for the computational workloads per host.

3.3 Energy Model

In this simulation model, the total energy consumption $engy_{tot}$ of a mobile host consists of five parts: $engy(SP)$, energy consumption of SP; $engy(EPIC)$, energy consumption of EPIC processor; $engy(LPIC)$, energy consumption of LPIC processor; $engy(xmit)$, energy consumption for transmitting data packet; $engy(rcv)$, energy consumption for receiving the data packet; and to satisfy

$$engy_{tot} = engy(SP) + engy(EPIC) + engy(LPIC) + engy(xmit) + engy(rcv)$$

3.4 Mobility Model

In our network model, to capture the randomly changing network topology, the position of each host is represented as a pair of coordinates (x, y) . The rates of movement, $dx(t)$ for the x coordinate and $dy(t)$ for the y coordinate, adhere to a given probability density function (pdf) $f_d(d(t))$ with mean $\mu_{d(t)}$ and standard deviation $\delta_{d(t)}$.

Dispatching a locally-generated request to a remote host results in energy consumption in the transceiver of the source host (and of course, energy consumption in the transceiver and possibly in the SP of the remote host.) In our simulations, we assumed that the energy consumption per data packet (for a fixed packet size) is proportional to the square of distance $D(t)$ between two mobile hosts. Denoting the sending energy for data packet by $g(D(t)^2)$, the total transmission energy can be calculated as:

$$engy(xmit) = \sum_{i=1}^n g(D(t)^2)$$

where n is the number of data packets that is transmitted per request.

3.5 Component Modeling

A. Model of the Local Service Requester

The local service requester in a mobile host i is modeled by a random variable LR_i indicating the local request generation rate, with a given pdf $f_{LR}(LR_i)$, with mean $LR\mu_i$, standard deviation $LR\delta_i$.

B. Model of the External Service Requester

The external service requester in mobile host i is characterized by a random variable ER_i indicating the external request generation rate, of which the pdf, $f_{ER}(ER_i)$, is defined as the convolution of the pdfs of two random variables, one indicating the external request generations coming from other hosts (type I external request) and the other indicating requests that were initially generated at the host itself, were sent off for remote processing, were rejected by other hosts, and have been returned to the host (type II external request.)

Consider that the random variable of the type I external requests for host i (denoted by $E1_i$) has a given pdf $f_{E1}(E1_i)$ with a mean of $E1\mu_i$ and standard deviation of $E1\delta_i$, and that the random variable of the type II external requests ($E2_i$) for host i has a pdf $f_{E2}(E2_i)$. The pdf of the time between request generation for external requests is defined as $f_{ER}(ER_i) = conv[f_{E1}(E1_i), f_{E2}(E2_i)]$. It is quite important to

note that the pdf of $E2_i$ is not known a priori but instead it is derived from the global state of network over the simulation time.

C. Model of the Local PIC

The local PIC in mobile host i (LP_i) is characterized by a random variable LP_i indicating its service time, with a given pdf $f_{LP}(LP_i)$, with mean $LP\mu_i$, and standard deviation $LP\delta_i$. The local PIC has two states: busy and idle. Upon receipt of a local request, if the local PIC is in idle state, the request will be processed by the PIC, otherwise, if the local PIC is busy, the request will wait in the local PIC queue. Let power consumption of the local PIC be denoted by $pow(LP)$, its switching energy by $swengy(LP)$, and its total energy consumption by $engy(LP)$, which can be calculated by:

$$engy(LP) = \sum_{s_j \in LPS} pow(s_j) \int_0^{T_S} \delta(s_j, t) dt + \sum_{\substack{s_i, s_j \in LPS \\ s_i \neq s_j}} swengy(s_i, s_j) \int_0^{T_S} \delta'(s_i, s_j, t) dt$$

where T_S denotes the total simulation time, the state set LP is $\{s_j | s_j = busy, s_2 = idle\}$, and

$$\delta(s_j, t) = \begin{cases} 1 & \text{if } LPS \text{ is designated as } s_j \text{ at time } t \\ 0 & \text{otherwise} \end{cases}$$

$$\delta'(s_i, s_j, t) = \begin{cases} \delta(t) & \text{if a transition from state } s_i \text{ to } s_j \text{ occurs at time } t \\ 0 & \text{otherwise} \end{cases}$$

D. Model of the External PIC

The external PIC in mobile host i (EP_i) is modeled as a random variable EP_i indicating the service time of PIC, with a given pdf $f_{EP}(EP_i)$, with the mean $EP\mu_i$, and the standard deviation $EP\delta_i$. Upon receipt of an external request, if it is a type II (already rejected) external request, then it will be sent to the SQ. If however, it is a type I (new) external request, it will be considered for further processing by the external PIC. In this case, if the external PIC is in idle state it will be processed by PIC, otherwise it will wait in the external PIC queue. Power of the external PIC is denoted by $pow(EP)$, its switching energy by $swengy(EP)$, and its total energy consumption by $engy(EP)$, which is calculated in a similar manner as that for the local PIC. Details are omitted due to the page limit.

E. Model of the Local PIC Queue

The local PIC queue in mobile host i is a FIFO-disciplined queue. The queue length is defined as random variable $LPQ_i = \{l_i : i=0, 1, 2, \dots, \mathcal{L}_i\}$, where \mathcal{L}_i is the maximum allowed queue length. The queue length increases as a new local request is generated, and decreases as a request leaves the local PIC. Upon receipt of a new request, if the queue length is equal to \mathcal{L}_i , the request will leave the host without any processing.

F. Model of the External PIC Queue

The external PIC queue in mobile host i is a FIFO disciplined queue whose length is defined by $EPQ_i = \{e_i : i=0, 1, 2, \dots, \mathcal{E}_i\}$, where \mathcal{E}_i is the maximum queue length. The queue length increases as a new external request is generated, and decreases as a request leaves the external PIC. Upon receipt of a new request, if the queue length is equal to \mathcal{E}_i , the request will leave the host without any processing.

G. Model of the SP

The SP in mobile host i (SP_i), is modeled to have the following state set $SSP_i = \{busy1, busy2, idle1, idle2, sleep\}$. The state transition of SP_i is depicted in **Figure 2**. SP_i in $busy1$ state processes workloads slower than when it is in $busy2$ state, but also consumes less power compared to when it is in $busy2$ state. The $idle1$ and $idle2$ states represent the state of the SP when it is up and running, but not servicing any request. Transitions between busy and idle states are autonomous and incur no cost whatsoever. When SP_i is in $sleep$ state it consumes very low power, but provides no service. The request service times for service provider i when it is in $busy1$ and $busy2$ states are random variables $ST1_i$ and $ST2_i$ with the associated pdf's of $f_{ST1}(ST1_i)$ and $f_{ST2}(ST2_i)$, respectively.

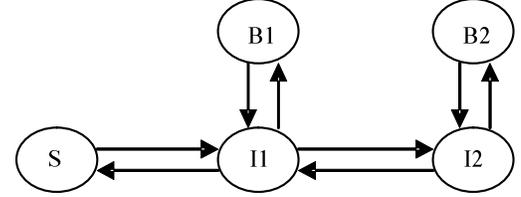


Figure 2. State transition diagram of the SP.

$$engy(SP) = \sum_{s_j \in SSP} pow(s_j) \int_0^{T_S} \delta(s_j) dt + \sum_{(s_j, s_k) \in TSP} swengy(s_j, s_k) \int_0^{T_S} \delta(s_j, s_k, t) dt$$

where $SSP = \{s_j : s_1 = busy1, s_2 = busy2, s_3 = idle1, s_4 = idle2, s_5 = sleep\}$,

$\delta(s_j, s_k, t)$ is a sampling function where an impulse appears at time t if a state transition from s_j to s_k occurs at this time instance. The set of all possible transitions is provided in the following state transition matrix of SP_i , for $s_j \neq s_k$ (a "1" in some cells means that the corresponding transition is possible.)

$s_j \backslash s_k$	1	2	3	4	5
1			1		
2				1	
3	1			1	1
4		1	1		
5		1			

Let set TSP denote the set of state pairs (s_j, s_k) that a transition from s_j to s_k , $s_j \neq s_k$, is possible and the $swengy(s_j, s_k)$ denote the switching energy for the state pair $(s_j, s_k) \in TSP$.

H. Model of the SQ

The SQ in mobile host i (SQ_i), is a combination of FIFO disciplined and priority queues of which the queue length is defined as $SPQ_i = \{s_i : i=0, 1, 2, \dots, \mathcal{S}_i\}$ where \mathcal{S}_i is the maximum queue length. Type II external requests have higher priority than the type I external requests and they line up in front of the SQ. The queue length increases as a request leaves any of the PICs or the host receives a type II external request, and decreases as the service provider starts servicing the request ejected from the front of the

queue. Upon receipt of a new request, if the queue length is equal to S_i , the request will leave the host without any processing (dropped).

I. Policy Evaluation

Due to the flexibility of NS, one can define a wide range of policies. In this paper we focus on the following different policies to illustrate the power and flexibility of the NS engine:

- Rejection policy for the type I external requests
- Send off policy for the local requests

The above-mentioned policies can be evaluated based on the following simulation conditions:

- Different \mathcal{L}_i , \mathcal{E}_i , and \mathcal{S}_i values for the maximum lengths of LQ_i , EQ_i , and SQ_i
- Different statistical distributions for request generation rates
- Different service rate distributions
- Different initial energies, initial locations and mobility rates for each mobile host.

4. An Energy-Aware Network Transaction Protocol (ENTP)

In this section, we propose an energy-aware network transaction protocol, which dynamically adjusts the computational workload among cooperative hosts to extend the network lifetime of a MANET.

For each mobile host, a cache table is maintained to record the remaining energy data of its adjacent hosts received during the last two communications and the related communication times. This data is used by the local host to predict the remaining energy of the adjacent hosts at the present time by using linear extrapolation method. Assuming the remaining energy of an adjacent host at times t_1 and t_2 are $remengy_1$ and $remengy_2$, respectively, the predicted remaining energy of this host at time t is given by:

$$remengy = remengy_2 + \frac{remengy_2 - remengy_1}{t_2 - t_1} (t - t_1).$$

The protocol works as follows:

- 1) When a host receives a local request, it evaluates its remaining energy, and looks up the cached table trying to find an adjacent host that has a larger predicted remaining energy than its own and the expected energy for migrating the request data is less than the energy required for its local execution. If such a host is found, it sends off the request to that host for remote processing. In addition, it sends along a message with its current level of remaining energy. Otherwise, it processes the request locally.
- 2) When a host receives a type I external request, the host records the attached remaining energy level about the sending host in its cache table and then compares it with its own current remaining energy. If the host has higher level of remaining energy than the sending host, it accepts the external request; otherwise it will reject the request by sending the request back with information about its own residual energy.
- 3) When a host receives a type II external request, the host assigns a high priority to the request, executes it locally, and updates the corresponding energy entry for the sending host in its cache table.

5. Experimental Results

In this section we present the results obtained by modeling a wireless MANET in the GPSS simulation environment. We built a simulation model to evaluate the performance of our proposed policy ENTP under different conditions and compared it with a no-send-off policy where no workload distribution is allowed. We performed two experiments for which the results are presented and analyzed in the following sections. In these experiments, we consider two types of average waiting time of the queues to be used as the network performance figures of merit. The average waiting time of all incoming service requests or of these requests that arrive when the related processor is busy, which we call “queued requests”. Due to the complexity of our model, the average waiting time cannot be determined analytically in either case.

5.1 Experimental Results of a Two-Host Network

In the first experiment, a two-host MANET model was set up. Based on the initial data summarized in **Table 1**, two different policies for redistribution of the workloads were considered and the performance of the network was evaluated under these conditions. The two network performance metrics were calculated by simulation. Figures 3 through 7 depict the results obtained by our experimentations.

The network energy utilizations of the two policies are compared in **Figure 3**. With the no-send-off policy, the energy resource of host 1 cannot be exploited to extend the lifetime of host 2 (which in this case has assumed the role of the network lifetime bottleneck), thus the overall network lifetime is short. A much longer network lifetime is achieved when using ENTP. From this figure, it can also be observed that when the difference between the initial battery energies of the hosts is between 10J and 35J, ENTP results in an almost 100% utilization of total energy. As the difference between the two initial energy levels goes outside this range, the energy utilization ratio drops however.

The increase in the total number of executed requests and the percentage of “exchanged” (migrated) requests, are related to the initial energy increase of host 1 in **Figure 4** and **Figure 5**, respectively. The origins of both plots correspond to the simulation results obtained with the no-send-off policy. When the initial energy increase of host 1 is relatively small, both the total number of executed requests and the percentage of exchanged requests increase almost linearly. However, when the initial energy of host 1 goes up, at some point, the total number of executed requests reaches its maximum value and stays unchanged afterwards. This phenomenon occurs because in such situations other parameters, such as SQ space of host 1 or request transmission energy of host 2, override the available energy of host 1 and become the critical factors that limit the redistribution of the computational workload between the two hosts.

In **Figure 6** and **Figure 7**, the average waiting time in the SQ of host 2 when employing the ENTP is reported for different conditions with variable initial energies for host 1 and variable service time for the SP of host 2. It is seen that when the service time of host 2 is relatively small, there is not much difference in the average waiting times even for very different initial energy levels for host 1. However, as the service time of host 2 increases, the average waiting time in the SQ of host 2 increases more slowly compared to the case in which host 1 has larger initial energy.

The proposed ENTP protocol significantly improves both metrics for the network performance compared to a policy in which

no workload distribution is allowed. As the difference between the initial battery energies of two mobile hosts increases, both metrics for the network performance increase until they reach their optimal value, and remain constant afterward. The optimal values of these metrics are different based on the differences in their battery energies and the rates of services. According to our simulation results, the optimum values of the network performance metrics are achieved at two distinct conditions. The first condition is when the energy difference increases by 23% and the service rate of the bottleneck host is increased by 35% and the second condition is when the energy difference increases by 22% and the service rate of the bottleneck host is increased by 5% (cf. **Figure 6** and **Figure 7**).

5.2 Experimental Results of a Multi-Host Network

For the second experiment, we constructed a ENS model of a MANET, which consists of 10 hosts. These hosts are free to move but are restricted to a 2000-by-2000 square foot area. The mobility rates of the hosts were assumed to be random variables with a Normal distribution with mean 0, where negative value means reverse direction. The local service request generation of each host was modeled with an exponential random value with the mean 600 ms that denotes the interval length between two consecutive requests. The service request generation rates (SRGR) are normally distributed over all hosts. Furthermore, to quantify the effects of mobility and variation of SRGRs on the network performance metrics, we assumed that all hosts are identical with the same initial battery energy capacity.

Figure 8 shows the network lifetime in terms of the *end-to-end throughput* (the number of requests that are serviced from the beginning of time until the first host dies due to lack of energy resource) for different mobility rates and SRGRs. This figure demonstrates that ENTP can increase the network lifetime by as much as 100%. It can also be observed that, in general, increase in the host mobility rate results in an increase in the *end-to-end throughput*. This is because, in a restricted area, high mobility causes the network topology to change fast, which in turn creates more opportunity for a heavily-loaded host to come close to lightly-loaded hosts with ample energies. As a result, the heavily loaded host can manage to distribute some of its workload among its neighboring hosts. Note however, that if the request generation rates of different hosts are nearly the same, then the best network performance will be achieved for lower host mobility rates. This is because there is less opportunity for workload distribution from one host to its neighbors (no one is doing any better or worse than any other; so each host will end up doing its own work).

The corresponding average task delay of the worst host during the network simulation is shown in **Figure 9** (for all requests) and **Figure 10** (for queued requests). From **Figure 9**, it is observed that the delay with an empty queue of ENTP is smaller than that of no-send-off policy when the standard deviation of SRGRs is larger than 20% of the mean value. As the variation increases, the ENTP delay decreases in terms of the percentage of the no-send-off policy delay. This is because the ENTP policy eases the workload of the hosts with lower residual energies (usually those with heavy loads) by migrating some of their tasks to the hosts with higher residual energies (usually those with light loads) and thus reduces the task delay for the worst hosts. However, in **Figure 10**, the task delay of ENTP with a non-empty queue is about 10% larger than that of the no-send-off policy except for the cases where the standard deviation is very large, i.e., 50% of the mean. The reason is that the ENTP policy encourages workload distribution to hosts with the

largest residual energies and thus may create bursts of workloads for these hosts, which in turn leads to higher task delays for type II external requests. The phenomenon of a larger delay seen in **Figure 9** for the case that the variation is very small, i.e. less than 10% of the mean, occurs because the effect of workload bursts overwhelms the benefit from workload migration from the heavily-loaded hosts. The workload burst effect may be reduced by setting a rejection threshold of SQ length for remote requests to a value smaller than the actual SQ size. This will however result in larger energy dissipation.

The simulation results of the network lifetime for different SP service times, which denote different ratios of computation energy to communication energy, are reported in **Figure 11**, where the standard deviation of SRGRs is assumed to be 40% of the mean. It can be seen that as this energy ratio increases, larger improvement on network lifetime may be obtained with ENTP.

6. Conclusion

In this paper we addressed the problem of analyzing dynamic power management policies in a wireless mobile ad hoc network with a dynamically changing topology. The non-Markovian stochastic nature of the problem, even in a simplified aspect of practical models, limits the use of analytical approaches. We developed a network simulation (ENS) model for evaluating power management policies for a MANET functioning in a real world environment. Furthermore, we developed an energy-aware network transaction protocol that dynamically redistributes computational workloads among the cooperative hosts within a MANET to achieve a better network lifetime. Using two measures of the network performances, the network lifetime characterized by the total number of the executed requests, before the host energies are exhausted and the average execution latency for requests, we evaluated the performance of the proposed protocol under a wide range of network conditions. Finally, by using ENS, empirical rules for redistribution of workloads and battery energy among the hosts in the network were derived and presented.

References

- [1] Brook S., *Computer Networks and Systems; Queuing Theory and Performance Evaluation*, Third Edition, Springer Verlag New York, Inc., 2000.
- [2] Douligeris C., and Mazumdar R., "A game-theoretic approach to flow control in an integrated environment," *Journal of the Franklin Institute*, 320(3), pp. 383-402, Mar. 1992.
- [3] Ferguson D., C. Nikolaou, and Y. Yemimi, "An economy for flow control in computer networks," *IEEE INFOCOM*, pp. 110-118, Apr. 1989.
- [4] Hass Z. J., "A new routing protocol for reconfigurable wireless networks," *ICUPC*, pp. 562-566, Oct. 1997.
- [5] Floyd S. and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, 1(4), pp. 397-413, Aug. 1993.
- [6] Hsiao M. T., and A. Lazar, "A game theoretic approach to decentralized flow control of markovian queue networks," *Performance*, pp. 55-73, Dec. 1987.
- [7] Karian, Z. A. and E. J. Dudewicz, *Modern Statistical, Systems, and GPSS Simulation*, First Edition, W. H. Freeman and Company, New York, 1991.

- [8] Lee S. J., M. Gerla, and C. K. Toh, "A simulation study of table-driven and on-demand routing protocols for mobile ad hoc networks," *IEEE Network*, pp.48-54, July 1999.
- [9] Maleki M., K. Dantu, and M. Pedram, "Lifetime prediction routing in mobile ad hoc networks," *IEEE Wireless Communication and Networking Conference*, Mar. 2003.
- [10] Royer E., and C. K. Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications.*, 6(2), pp. 46-55, Apr. 1999.
- [11] Scott K., and N. Bombos, "Routing and channel assignment for low power transmission in PCS," *ICUPC*, vol. 2, pp. 498-502, Oct. 1996.
- [12] Shenker S., "Making greed work in networks; A game theoretic analysis of switch service disciplines," *IEEE/ACM Transactions on Networking*, pp. 819-831, Dec. 1995.
- [13] J. Short, R. Bagrodia, and L. Kleinrock, "Mobile wireless network system simulation," *Wireless Networks*, pp. 451-67, 1995.
- [14] Singh S., M. Woo, and C.S. Raghavendra, "Power aware routing in mobile ad hoc networks," *IEEE/ACM MOBICOM*, pp. 181-190, Oct. 1998.
- [15] C. K. Toh, "Associativity based routing for ad hoc mobile networks," *Wireless Personal Communications Journal*, pp. 103-139, Mar. 1997.
- [16] C. K. Toh, "Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks," *IEEE Communications Magazine*, pp. 138-147, June 2001.
- [17] J.L. da Silva, J. Shamberger, M.J Ammer, etc, "Design methodology for PicoRadio networks," *DATE 2001*, pp. 314 -323, 2001
- [18] L. Shang, R. P. Dick and N. K. Jha, "An economics-based power-aware protocol for computation distribution in mobile ad-hoc networks," *IASTED PDCS*, Nov. 2002

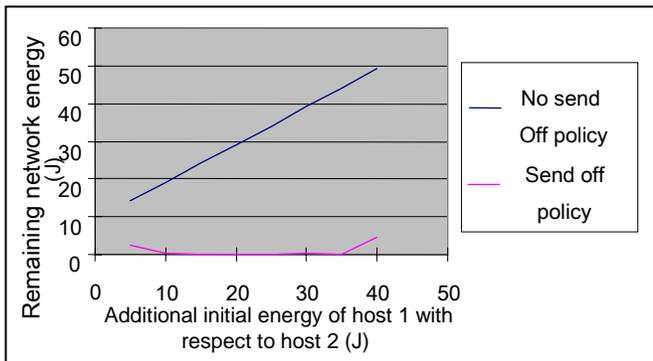


Figure 3. Network energy utilization.

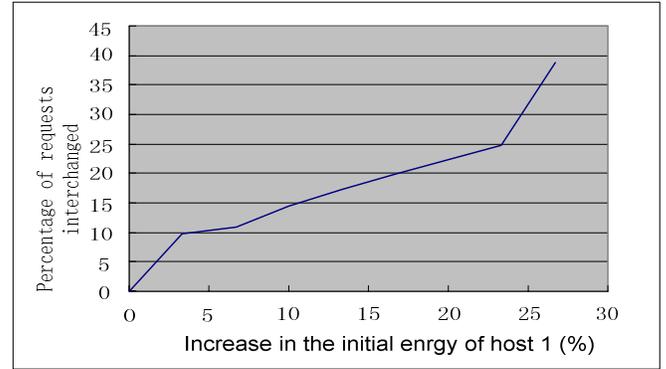


Figure 4. Percentage of requests interchanged between the two hosts.

Table 1. Initial condition of network model

Local & external request generators for the two hosts	Poisson with requests average arrival rate of 1/60 ms
Local & external PIC service times for the two hosts	Exponential with average service time of 4 ms
Service time of the service provider, in the high speed state, in the two hosts	Exponential with average service time of 7 ms
Service time of the service provider, in the low speed state, for the two hosts	Exponential with average service time of 12 ms
The distance between to hosts	Initially at 200ft, changed based on the Normal distribution with the speed of 10 ft per second
Initial battery energy, host 1	155,000 mJ
Initial battery energy, host 2	150,000 mJ
PIC power dissipation, host 1	Local 0.10 W, external 0.09 W
PIC Power dissipation, host 2	Local 0.11 W, external 0.10 W
SP power dissipation, host 1	<i>busy1</i> : 0.125 W, <i>busy2</i> : 0.500 W, <i>idle</i> : 0.125 W, <i>sleep</i> : 0.066 mW
SP power dissipation, host 2	<i>busy1</i> : 0.100 W, <i>busy2</i> : 0.400 W, <i>idle</i> : 0.125 W, <i>sleep</i> : 0.066 mW
SP switching energy, host 1 & 2	<i>busy1</i> to <i>idle1</i> , vice versa: 0 <i>busy2</i> to <i>idle2</i> , vice versa: 0 <i>idle1</i> to <i>idle2</i> , vice versa: 0.05 mJ <i>idle1</i> to <i>sleep</i> , vice versa: 0.20 mJ
Maximum SQ length	8 requests
SQ length threshold for switching from low speed to high speed	4 requests

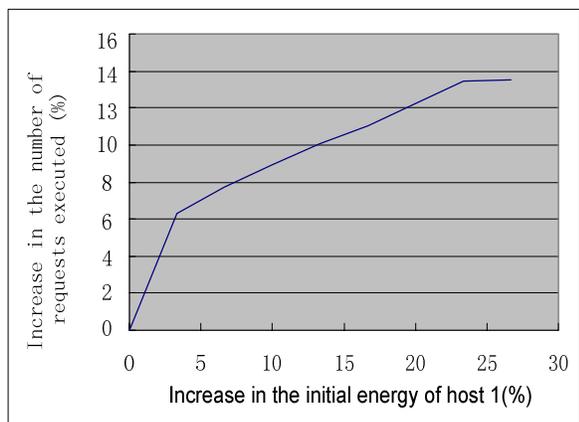


Figure 5. Throughput increase of the two-host network.

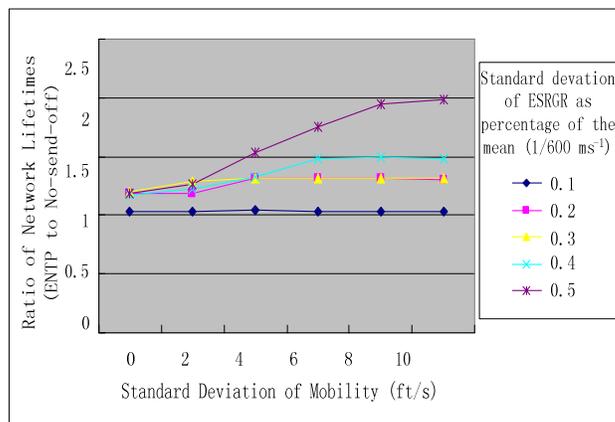


Figure 8. Increase in network lifetime when using ENTP.

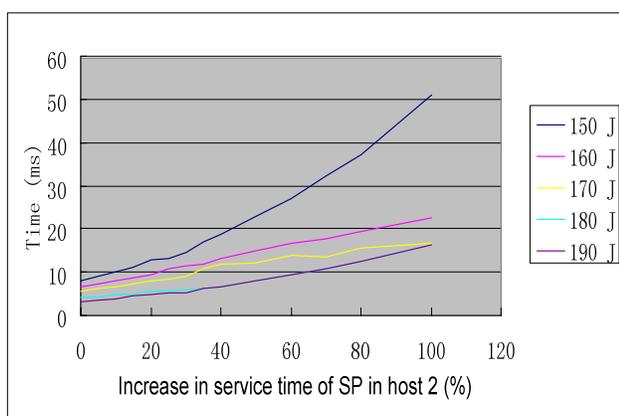


Figure 6. Average waiting time in SQ of host2 (all requests).

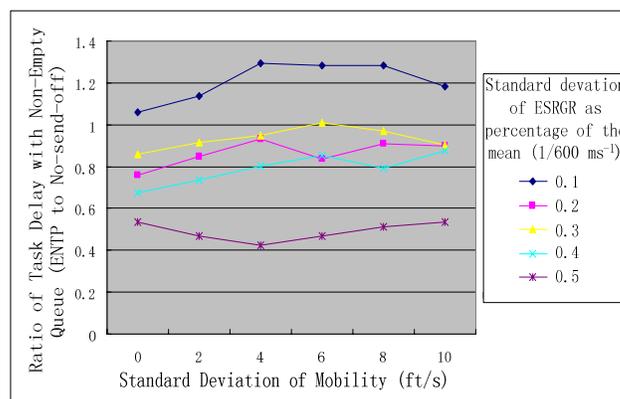


Figure 9. ENTP: Average waiting time of the worst host (all requests).

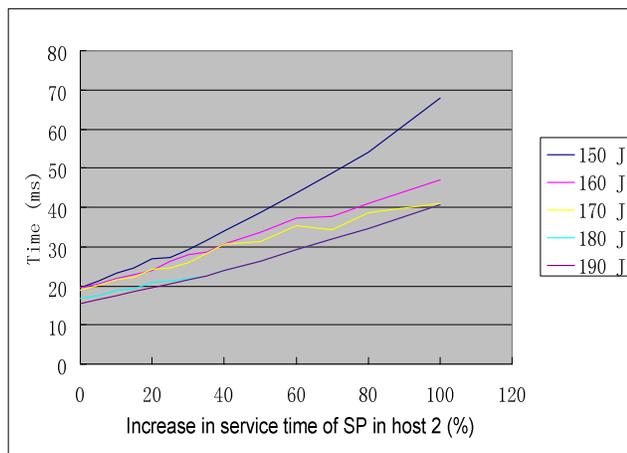


Figure 7. Average waiting time in SQ of host2 (queued requests).

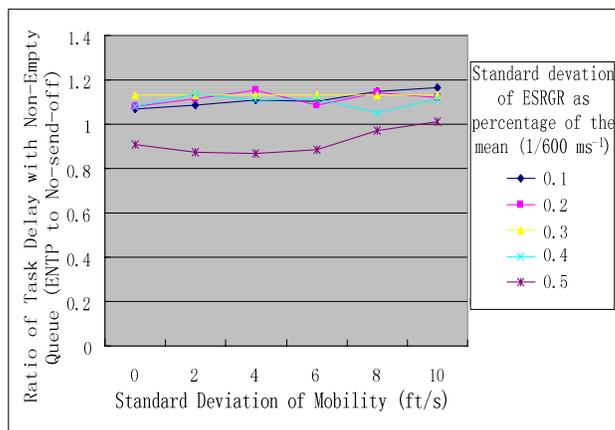


Figure 10. ENTP: Average waiting time of the worst host (queued requests).

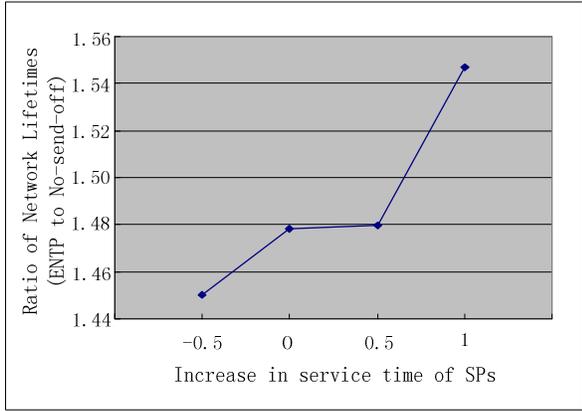


Figure 11. Network lifetime increase vs. SP service time.