

# Optimizing Power Aware Routing in Mobile Ad Hoc Networks\*

Jharna Chokhawala and Albert Mo Kim Cheng  
Real-Time Systems Laboratory  
Department of Computer Science  
The University of Houston-University Park  
Houston, TX 77204-3010,U.S.A  
jharna@cs.uh.edu,cheng@cs.uh.edu

## Abstract

The chief limiting factor for current mobile devices is the amount of battery power. Most of these devices run on lithium-ion, rechargeable batteries. These batteries have a lifetime of a few hours of active workload and about 1-2 days of idle time. To improve this crucial factor, researchers have tried to optimize power consumption in every aspect of the mobile device. Power consumption can be optimized by disks, memory chips, CPU scheduling, applications and communication techniques. In this project we try to optimize a power efficient routing technique for ad hoc networks, proposed by Ivan Stojmenovic and Xu Lin in [3]. They have implemented a routing algorithm, which tries to minimize the power consumed in transmitting a packet from the source to the destination while trying to maximize the lifetime of the network by avoiding nodes that have a shorter lifetime remaining. We try to optimize this method by introducing a threshold on the remaining lifetime of a node. This threshold indicates whether the node should be included in making routing decisions for a packet. Initial results indicate that this threshold depends on various factors and can be adjusted for each network.

## 1 Introduction

The types of mobile devices and their applications have grown exponentially over the last few years. They range from laptops, PDA's, notebooks to cell phones. Most of these devices can currently perform all the tasks of a traditional PC with the advantage of portability. However, they are limited in the duration of activity that can be accomplished during the lifetime of their batteries. Lithium-ion rechargeable batteries are the most commonly used batteries. They have a typical lifetime of a few hours of active workload and a couple of days of idle time.

Mobile devices consume power even in their sleep modes. For example, in mobile

phones, even if they are not in use, there is a constant power drain because the trans-receiver is constantly hearing for signals to itself. In sleep mode, the power consumption (Wavelan Metricom and IR) ranges between 150-170mW, while in idle state the power consumption goes up by one order of magnitude. In transmit mode the power consumption is almost doubled [4].

In this project, we concentrate on ad-hoc networks. Ad-hoc networks are formed where there is no existing infrastructure and there is a need for communication. Examples of ad hoc networks include soldiers on enemy terrain, workers in a disaster area, or a group of executives at an outdoor location. Figure 1 shows a typical ad hoc network.

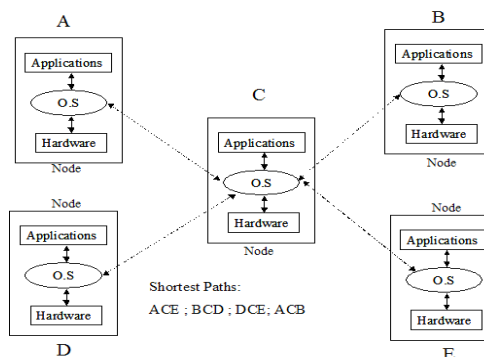


Figure 1. Ad hoc networks

What differentiates ad hoc networks from traditional wireless networks is the absence of a centralized base station. In traditional wireless networks, nodes wishing to communicate with each other, have to first contact the nearest base station, which forwards their requests to the base station closest to the destination node. All packets are routed through the path established by the base station. The base stations perform the tasks of tracking, routing and route maintenance. In ad hoc networks, all these tasks are performed by the nodes themselves, in addition to their personal tasks. This causes additional drain on the batteries leading to a diminished lifetime. Power utilization can be optimized by employing

\*This material is supported in part by a grant from the Institute of Space Systems Operations.

routing algorithms, that avoid nodes with less battery power remaining while trying to minimize the total power consumed in transmitting a packet. In this project we have tried to optimize the localized power aware routing algorithm proposed in [3].

The remainder of this paper is organized as follows. Section 2 describes the various power saving techniques employed by mobile operating systems and devices. Section 3 describes various power aware routing algorithms. Section 4 describes the modified algorithm, methods and results. Section 5 gives the conclusion and future work.

## 2 Power saving techniques

A lot of effort and research is currently on going to reduce the power consumed in each and every aspect of a mobile device. We give a brief description of some of the methods in the following sections.

### 2.1 Disk Scheduling

One method of energy conservation is to spindown a disk in its idle time. The spindown delay is the amount of time the disk is idle before it spins down. [2] presents a quantitative analysis of the potential costs and benefits of spinning down a disk in its idle time. The tests were carried out using traces from both DOS machines and the Sprite File system. The conclusion was, that the maximum power savings were obtained by using a spindown delay of 2 seconds as opposed to the 3-5 minutes recommended by most manufacturers.

To justify this claim, the authors presented 2 points: frequency of sleep and length of sleep. They claim that, with shorter delays, the disk gets to sleep for a longer time and hence save more power.

The drawback of spinning down a disk after such short delays is the time and energy needed to spinup the disk, which results in user delay. Traces used by the authors show that the spindown occurs 8-15 times an hour. This translates to 16-30 seconds of user delay per hour, which is reasonable compared to the power savings incurred.

### 2.2 CPU Scheduling

The power consumed by a processor is directly proportional to the supply voltage, the

switching capacitance of the various devices and the frequency of the clock. Gates in CMOS CPU's switch state at every clock cycle, which lead to a short circuit between the power-supply and ground. As a result more power is wasted with higher frequency.

The power required by the CPU is given by  $CV^2F$ , where C is the total capacitance of the wires, V is the supply voltage and F is the operating frequency. There are various algorithms proposed for adjusting the clock frequency in idle time. The main idea behind it is to balance the CPU usage between bursts of high utilization and idle times. Task or process scheduling can be an effective way of accomplishing this.

Almost all processes have a deadline by which they need to be executed. It has been observed in [1] that even when the processor is operating at the worst case, in scheduling the tasks, there is some idle time. This idle time is called the slack time. This slack time can be used to conserve energy by slowing down the processor and reducing the voltage. These techniques are known as, static slowdown and voltage scaling. We can reduce or eliminate the idle time by reducing the voltage to operate the processor such that, the process takes longer to finish but is completed before its deadline.

### 2.3 Memory Allocation

In mobile devices, memory instructions are among the highest consumers of power [2]. Since many small devices do not have a secondary storage, the power consumed by the memory is very crucial and needs to be optimized. Direct Rambus DRAM (RDRAM), have come out with a DRAM that allows the individual devices to be in different power states. These devices are in decreasing order of power states and increasing order of access times: Active, Standby, Nap and Powerdown. There are 2 types of hardware policies for determining the states of the memory chip.

- Static: Every power-aware DRAM chip is in the same power mode when there are no outstanding requests for that device.
- Dynamic: In this strategy, the OS tries to predict the requests based on the access patterns and access times to power down the chips to a lower state.

Placement policies for code and data can also help reduce power consumption. If active pages with temporal locality are grouped together and

placed on the same memory chip before moving to the next, the remaining chips can be powered down [2]. This technique helps in reducing the power consumed in reading data from memory. The simulation results given in [2] show power saving of about 6% - 50% using the static, dynamic and temporal locality placement policies.

### 3 Power aware routing algorithm

This research deals mainly with localized algorithms. In localized algorithms, the nodes in the network make routing decisions based solely on the location of itself, the location of the destination and the location of its neighbors [3]. Localized algorithms are distributed algorithms where simple local node behavior achieves a desired global objective [3]. Non-localized algorithms are those in which the nodes require the complete knowledge of all the nodes in the network along with the corresponding edges. In ad hoc mobile networks, nodes are moving at all times and there may be several nodes exiting and entering the network at any given point of time. To keep a track of all these nodes and their corresponding edges is cumbersome and requires a huge overhead. To avoid this overhead, routing decisions are made on demand using the dynamic source routing technique proposed in [5].

Intuitively, if we want to minimize the total power required in transmission, the shortest path would be the optimal solution. This is not always true.

In fig. 1, node C is on the shortest path for a number of source – destination pairs. It will get depleted the fastest and thus lead to a breakdown of the network. To avoid this condition, the remaining battery power of each node needs to be taken into consideration. This is known as cost-aware routing.

We now introduce some of the existing power aware metrics and routing algorithms.

#### 3.1 Existing Power Aware Metrics and Routing Algorithms

There are a number of power and cost aware metrics present. The two basic ones are :

- Power aware routing: In this case, the transmission power depends on the distance between the source and the destination.

- Cost aware routing: In this case, the routing decisions are made based on the remaining life-time of nodes between the source and the destination.

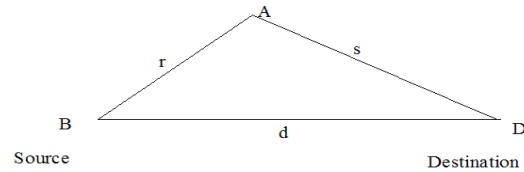


Figure 2.

#### Combining Cost and Power[3]

Let A be a neighbor of B (node currently holding the message) that minimizes  $pc(B,A) = \text{power-cost}(B,A) + v(s)f'(A)$   
Where  $s = 0$  for D is D is a neighbor of B

This project has specified 2 ways of combining power and cost

- Power \* Cost  
 $\text{Power-Cost}(B,A) = f(A) * u(r)$  where  $|AB| = r$
- Power + Cost  
 $\text{Power-Cost}(B,A) = \alpha u(r) + \beta f(A)$  where  $\alpha = f'(s)$  and  $\beta = u(r')$   
 $r' = \text{average length of all edges going out of } s$

$$\text{Power-Cost}(B,A) = f'(s) + u(r')f(A)$$

#### Power-Cost efficient algorithm[3]

##### Algorithm

Power-cost routing (S,D)

A := S;

Repeat

B := A;

Let A be neighbor of B that minimizes

$pc(B,A) = \text{power-cost}(B,A) + v(s)f'(A)$ ;

Send message to A

Until A = D (\* Destination reached \*)

or A = B (\* Delivery failed \*);

### 4 Methods and Results

In this section, we describe the proposed algorithm and the parameters considered for conducting this experiment. We extend the power-cost efficient algorithm described in the previous section to implement timing constraints. The results of the power-cost aware algorithm (given in the next section) show that it performs better when the network/graph is dense. In a large network, a node will have a

large number of neighbors. The computation time for calculating the minimum power-cost among the nodes' neighbors is quadratic or exponential (depending on the algorithm used, power+cost or power\*cost). In order to reduce this computational time we introduce a threshold value for the remaining battery power of the nodes. While selecting a route, nodes with battery power greater than the threshold will only be considered. It would then go on to compute the minimum power-cost route. However, if none of the nodes meet the threshold, the threshold is reduced by half. This will continue until a node meeting the threshold is found or the threshold reduces to a minimum specified value. This would imply that the network is broken and the packet cannot be delivered. An appropriate error message is then given. The modified algorithm is as follows:

Threshold = 50%; success = 0; cutoff = 10%

A := S;

**Repeat**

  If  $g(A) \geq \text{threshold}$  then

    B := A;

    Let A be neighbor of B that

minimizes

$$pc(B,A) = \text{power-cost}(B,A) + v(s)f'(A);$$

    Send message to A;

    success = 1;

**Until** A = D (\* Destination reached \*)

  or if success < 1 then

    if threshold > cutoff then

      threshold = threshold / 2;

    or A = B (\* Delivery failed \*);

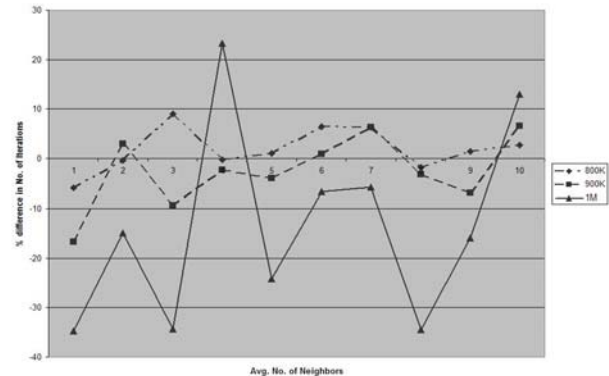
For this project, we simulated an ad-hoc network with varying densities and transmission ranges. Each node had a random location in a 1000 x 1000 area. They also had a random amount of initial power within a range of 700k to 1500K. The thresholds were varied from 700k to 1000K.

The initial results are shown in graph 1.

As seen from the graph, the life of the network depends on various factors, which include the initial threshold, density of the network and range of the transmitters. These parameters need to be adjusted for each network. This is still work in progress.

## 5 Conclusion and Future Work

In this project, we have studied the various power saving techniques employed by mobile devices. We have simulated an ad hoc network and implemented the Power-Cost algorithm proposed by I.Stojmenovic and X. Lin



Graph 1. The no. of iterations possible with varying thresholds

in [3] along with our modified algorithm. The initial results show that the various parameters affect the lifetime of the network. This is still work in progress. At this point, we are not taking into consideration the nodes' personal tasks. In the future, we can take into account the power consumed by these activities, to see its impact on the lifetime of the network.

## 6 References

- [1] F. Li, R. Sutton, J. Rabaey, "Low Power Operating System for Heterogeneous Wireless Communication Systems", Workshop on Compilers and Operating Systems for Low Power 2001, September, 2001
- [2] V. Tiwari, S. Malik, A. Wolfe, "Power analysis of embedded software: A first step towards software power minimizations". IEEE Transactions on Very Large Scale Integration, 2(4): 437- 445, December 1994.
- [3] I. Stojmenovic, X. Lin, "Power-aware Localized Routing in Wireless Networks", IEEE Transactions on Parallel and Distributed Systems, Vol. 12, no. 11, November 2001.
- [4] M. Stemm, P. Gauthier, D. Harda, Reducing Power Consumption of Network Interfaces in Hand-held Devices,
- [5] D. B. Johnson, D.A. Malt, "Dynamic source routing in Ad Hoc Wireless Networks", in Mobile Computing, T. Imelenski and Hank Korth, Eds. Kulwer Academic Publishers, 1994.