

Power Mode Scheduling for Ad Hoc Networks

Santashil PalChaudhuri and David B. Johnson
Rice University
Department of Computer Science
santa@cs.rice.edu, dbj@cs.rice.edu

Abstract

An ad hoc network is a group of mobile wireless nodes that cooperatively form a network among themselves without any fixed infrastructure. Increasingly, power consumption within ad hoc networks is becoming a core issue for these low-power mobile devices. This paper focuses on a novel approach for energy conservation within the routing protocol of the ad hoc network. A wireless network interface in sleep mode expends an order of magnitude less power than in idle mode, but no packets can be sent or received while in sleep mode. In this paper, we propose two probabilistic algorithms, for scheduling transition from idle mode to sleep mode. Performance evaluation of these strategies show a substantial reduction in power usage, with only a slight decrease in performance.

I. Introduction

Energy consumption in mobile computing devices is universally recognized as an important issue for further research. The sources of power consumption are communication and computation, with communication often being the chief power consumer.

An ad hoc network is a group of mobile wireless nodes that cooperatively form a network among themselves without any fixed infrastructure. Each node in the ad hoc network forwards packets for other nodes, to allow nodes not within direct wireless transmission range to communicate. There has been considerable research on conserving power in these routing protocols, although most of this research has focused on controlling the transmission power of the sender's network interface. Although significant in terms of reducing the power consumption in the wireless transmitter of a sender, it does little to conserve power among the other nodes — receivers, forwarders, and nodes not involved in this communication.

The network interface hardware at a receiver node can operate in any of four different modes:

- *Transmit mode*: The mode at a node when transmitting a packet.
- *Receive mode*: The mode at a node when receiving a packet.
- *Idle mode*: The mode generally used at a node with the node is neither transmitting nor receiving a packet. This mode consumes power because the node has to listen to the wireless medium continuously in order to detect a packet that it should receive, so that the node can then switch into receive mode.
- *Sleep mode*: Sleep mode has very low power consumption. The network interface at a node in sleep mode can neither transmit nor receive packets; the network interface must be woken up to idle mode first by an explicit instruction from the node.

In the evaluation of any algorithm for energy conservation, an estimate of energy consumption is necessary.

Feeney [1] shows the specification and actual measured current drawn by one popular wireless network interface card in the four possible modes. Receive and idle mode require similar power, and transmit mode requires slightly greater power. Sleep mode requires more than an order of magnitude less power than idle mode. These measurements show that the network interface expends similar energy, whether it is just listening or actually receiving data. Hence, intelligently switching to sleep mode whenever possible will generally create significant energy savings. We now present two probabilistic algorithms which will achieve this. The full version of these protocols is available from the thesis [2].

II. Algorithm for Sleep-Mode Scheduling

All of the nodes in an ad hoc network do not always participate in receiving, sending, and forwarding of data packets. Rather, many nodes do not need to be logically in the topology to maintain connectivity. If these nodes can be

identified, then they can be put into sleep mode rather than staying in idle mode.

IEEE 802.11 Power Saving mode works only in the case where all the nodes are reachable in a single hop from every other node. Our two algorithms are based on this, and extend it to include multi-hop ad hoc networks. We assume clock synchronization between the nodes through some mechanism.

II-A The Adaptive Sleep Algorithm

In the Adaptive Sleep Algorithm, a node periodically switches between idle and sleep modes.

A interval called *Sleep Interval (SI)* is a period of time from the last the receipt of any Route Request by a node. After SI time interval, the *periodic sleep* cycle starts. The periodic sleep cycle repeats at a period called the *Sleep Period (SP)*. It also has a variable *Duty Cycle (D)*, which is the fraction of the SP period that the node is in idle mode. For $(1 - D)$ fraction of SP, the node remains in sleep mode. Anytime a sender in sleep mode needs to send data, it transitions to active mode and starts sending. When a node is in sleep mode, if a ROUTE REQUEST is transmitted by a neighbor node, it will not be able to receive it. Depending on the length of time that the node is in sleep mode, it might wake up in time to receive the subsequent re-broadcasts of the ROUTE REQUEST messages by the node's neighbours. Hence, the SP needs to be small enough so that, although a node may have missed the initial ROUTE REQUEST, it wakes up in time to receive the subsequent re-broadcasts.

In densely populated ad hoc networks, many nodes can be interchangeably used for routing purposes. In a dense network, any node might sleep for a longer duration, as the relative utility (for forwarding purposes) of all nodes decrease.

1. *Estimate Neighbour Count:* There are a number of published approaches for estimating the neighbour count of a node in a wireless network.
2. *Estimate of Duty Cycle:* The idea is to keep the total idle time for all nodes in a neighbourhood constant. So, the duty cycle is kept proportional to the recent estimate of neighbour count.

There is no need for accurate count of the number of neighbours, since the algorithm provides a natural feedback mechanism. If the neighbor count is underestimated by a node, that node stays awake for a longer time, during which it hears from more neighbours, and subsequently reduces its duty cycle.

II-B. Birthday Sleep Algorithm

The inspiration behind this protocol is the *birthday paradox*, which is the probability that at least two people in a room have their birthday on the same day. We use a similar idea for convergence between a receiver and a transmitter.

Suppose that over a period of n fixed length slot periods, two wireless nodes independently and randomly select m of these slots. In each of these m slots chosen by a node, the node remains in idle mode listening for packets. In the remaining $n - m$ slots, it goes to sleep mode, thereby saving energy. The energy saved is approximately $(n - m)/n \times 100$ percent of the original energy consumed by the wireless network interface.

For two nodes, A & B: The probability of B choosing k slots all different than those chosen by A is

$$C(n - m, k)/C(n, k) \quad (1)$$

The probability of B choosing at least 1 slot the same as those chosen by A, within k slots is

$$1 - C(n - m, k)/C(n, k) \quad (2)$$

Let $p_i = m_i/n$ denote the probability of staying awake for $node_i$, where m_i is the number of slots $node_i$ is awake and n is the total number of slots.

If a node is a sender or receiver of data or if it is an active forwarder of data, then the node always remains awake. Hence,

$$p_i = 1 \text{ for } i \in S, D, F$$

where S is the set of all active source nodes, D is the set of all active destination nodes, and F is the set of all active forwarder nodes at the current time. For all other nodes, p_i depends on the policy.

All nodes are awake for the first slot out of n slots. Depending on p_i , each node randomly chooses m_i out of the n slots to be idle. Each node builds a *Sleep Indication Map (SIM)* containing n bits, with the r^{th} bit set to 1 or 0 depending on whether the node will be idle or asleep in the r^{th} slot of that T period. A node uses the knowledge gained from the SIM packets from its neighbors to build a *Sleep Indication Table (SIT)*.

A node uses its SIT to schedule packets to any neighbor.

References

- [1] L. Feeney and M. Nilsson. Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment. In *Proceedings of INFOCOM 2001*, volume 3, pages 1548–1557, Anchorage, Alaska, Apr. 2001.
- [2] S. PalChaudhuri. Power Mode Scheduling for Ad Hoc Network Routing. Masters Thesis, Computer Science, Rice University, May 2002.