SLEEP MODE OPERATION OF A ROUTING PROTOCOL IN MOBILE AD HOC NETWORKS

A. QAYYUM
Avaz Networks, 5-A Constitution Avenue, Islamabad, Pakistan
E-mail: qayyum@avaznet.com

Y. TOOR
NUST - MCS, Humayun Road, Rawalpindi, Pakistan
E-mail: yassertoor@hotmail.com

P. JACQUET, P. MÜHLETHALER AND T. CLAUSEN
INRIA Rocquencourt, B.P. 105, 78153 Le Chesnay Cedex, France
E-mail: [Philippe.Jacquet, Paul.Muhlethaler, Thomas.Clausen]@inria.fr

This paper presents an algorithm for power conservation that can be used with OLSR, a proactive routing protocol. The algorithm enables mobile nodes to go into sleep mode to conserve their battery resources. The algorithm keeps the sleep mode operation of a node only known to its immediate neighborhood by storing the traffic destined for a sleeping node in its immediate neighborhood. The mechanism also tries to avoid, if possible, the partitioning of network caused by sleep mode of a node that links the partitions. A simulation study is also presented which evaluates the performance of a network using this scheme in terms of network lifetime, average packet delay, etc.

1 Introduction

Current research has made much progress in increasing the battery resources of small mobile devices. However, it also resulted in increased usage of these resources following the supply-demand phenomenon. Therefore, the challenge of economically using the limited resources is still there. Battery resources are paramount to the operational lifetime of a small mobile terminal and hence an efficient use of them is highly desirable. Many power conservation techniques focus the system hardware at physical layer while other approaches target more energy aware protocols which use the system modules efficiently.

Most energy conserving protocols are developed for a centralized environment. The central node or base station buffers the traffic destined for the mobile devices thus allowing the mobile devices to enter into a low power or sleep state. They have to wake up after a predetermined interval and collect the packets from the base station. These strategies are not suitable for ad hoc networks where there exist no such central entity.

1.1 Ad hoc network scenarios

In a structured wireless network, power conservation techniques may be usefully employed if a node does not receive, or stop receiving as early as possible, the traffic not destined to itself. However, an ad hoc network presents a difficult problem where every node overhears any transmission in its radio range. It is required because the nodes themselves provide the routing infrastructure. A node which is completely inactive at the application layer may in fact be essential in maintaining network connectivity. Therefore application-level power saving techniques are not advisable.

A communication protocol may help to conserve the battery by introducing silent or sleep periods in its operation. However, these sleep intervals may not be used unnecessarily, especially in disperse ad hoc networks where routing of packets is highly dependent on the availability of intermediate nodes. In these situations, a sleep mode operation of a node may disconnect some nodes from the network. A balance between the number of sleepy and active nodes is important in order to efficiently use the network resources.

1.2 Power consumption in wireless interfaces

Wireless network interface may consume a significant part of energy of a small mobile device [1]. In a wireless interface, power consumed during transmission of packets is significant as compared to reception. However, if a broadcast message is received by several neighbors, the total cost of receiving the message becomes greater than the cost of transmission [2]. In a network, time spent in transmission and reception by a node is usually small as compared to that in idle mode. However, the power consumed in idle mode is not significantly low as might be thought [3].

A power conservation technique that requires to change the state of the network interface card may be implemented by modifying the device drivers. The power management features of a network card are not fully utilized in the drivers. PC card standard defines a few configuration registers in a card's memory space. One could write to these registers to signal that the card should enter into a power down state. Alternatively, functions provided by the Linux PCMCIA Card Services interface could be called by a device driver to power down or power up a card device [4].

In this paper, we start by reviewing some approaches of power conservation which are proposed for MAC or higher layer protocols. In the next section, we describe our scheme of power conservation that is integrated with the OLSR protocol. Finally, we discuss the simulation results of the proposed scheme and end up with conclusions.

2 Power conservation techniques

Many schemes follow the approach of decreasing the signal power level to a minimum, that is required for a successful transmission. But these schemes require interaction with physical layer to change the transmitted signal level. Some techniques use the routing protocols to calculate the routes which result in consuming the minimum transmission power. This approach assumes that different power transmission levels may be used for different transmitter-receiver pairs, and a routing protocol knows the transmission levels for each of these pairs in a network.

Another approach is to allow some nodes to become inactive for communication for a small interval. These techniques are mostly implemented at the MAC layer but few protocols also propose it at the network layer. Our study relates to power conservation at the network layer and follows this approach of sleep mode, or inactive intervals. We briefly discuss below some techniques which are related to sleep and wake period approach.

2.1 Common standards

GSM (Global System for Mobile Communication) uses Discontinuous Transmission (DTX) [5] in which transmitter is turned off during silence periods in speech. In HIPERLAN (High Performance Radio Local Area Network), a node is allowed to go into inactive state for a short interval and its neighboring power supporter nodes communicate with it only when it is active and buffer its incoming packets while it is not active.

In Bluetooth, the nodes which are not addressed in a packet can go to sleep for remaining slots of the packet. In its Sniff mode, a node participates in the piconet only in sniff interval. In Hold mode, no data is transmitted for a hold mode length in order to conserve power. In IEEE 802.11b, nodes are synchronized and power up at a predetermined time in the ad hoc mode. The nodes which have no data to receive return back to power save mode. In its client/server mode, the Access Point (AP) buffers the packets of a power saving node which retrieves its data from the AP when it wakes up.

2.2 PAMAS

PAMAS (Power Aware Multi-Access) protocol [6] for ad hoc networks modifies the MACA protocol by providing separate channels for control (RTS/CTS) and data packets. A node with data packet to transmit sends an RTS (Request-to-send) and awaits the CTS (Clear-to-send) from the receiver. If it receives a CTS then it sends the data packet over the data channel, otherwise, it enters a backoff (or sleep) state. The receiver node sends a “busy tone” over the control channel to let other nodes know that the data channel is busy. Hence, these nodes may turn off their wireless interfaces.

2.3 BECA and AFECA

Basic Energy Conserving Algorithm (BECA) [7] is designed to run on top of routing protocols. The algorithm allows energy constrained devices to enter into sleep mode by turning off their radios. A device turns on the radio after a specific duty cycle and listens. If no packet arrives in a specified interval the device goes back to sleep. Because a route request packet may arrive while a node is asleep, the algorithm requires the routing protocol to repeat the request a specific number of times, in case it does not receive a reply.

Adaptive Fidelity Energy Conserving Algorithm (AFECA) [7] suggests that sleep period of a device can be longer in denser networks since alternative routes can be easily computed. A device can determine the number of its neighbor nodes. Higher the number is, longer is the sleep period of the node. AFECA shows a slightly more reduction in power consumption than BECA.

2.4 LEAR

Local Energy-Aware Routing (LEAR) protocol [8] tries to achieve balanced energy consumption among all nodes. A node relies on its remaining battery level to decide whether or not to be a part of routing paths. An energy hungry node can conserve its battery power by not forwarding packets on behalf of other nodes. This approach is close to sleep mode in the sense that a node declares itself unavailable for routing of packets when its battery level goes down. However, the energy hungry node keeps listening its own packets.

3 Power conservation in OLSR

Several techniques exist for wireless networks which use the notion of sleeping nodes to implement a power conservation scheme. However, a sleeping node may be considered as an unreachable node by a routing protocol. This may result in discarding the traffic of sleeping node. We focus our study on this issue, and develop a sleep mode mechanism that may work with a routing protocol. This coupling of sleep mode with the routing protocol keeps the connectivity of the sleeping node with the rest of the network. This approach helps a node to have a short sleep period without being considered unreachable. The scheme we developed works with OLSR protocol.

3.1 OLSR protocol overview

Optimized Link State Routing (OLSR) [9] is a proactive routing protocol, based on periodic exchange of control messages. Two basic types of control messages in OLSR are Hello and Topology Control (TC) messages. Hello messages are broadcast to one-hop neighbors only, and contain the list of known neighbors along with their link status. In this way, Hello messages enable a node to acquire information up to two-hop neighbors. Based upon this information, a subset of one hop neighbors is selected, called Multipoint Relays (*MPRs*), which covers (or give access to) the entire two-hop neighbor set. A node indicates its current MPR neighbor nodes in its Hello messages with the link status "MPR". These MPRs forward broadcast messages of the selecting node during a flooding process. The MPR technique significantly reduces the number of retransmissions in a flooding process [10,11].

The neighbors of a node which select it as an MPR are called *MPR Selectors* of that node. A node periodically disseminates the information about its current *MPR Selectors*, in the entire network through TC messages. This information is used by the nodes for route calculation. As a consequence hereof, the routes contain only MPRs as intermediate nodes, i.e., all the intermediate nodes along the path are selected as an MPR by at least one of their neighbors. It implies that as long as a node is present in an *MPR Selector* set of any of its neighbor nodes, its presence is announced in the network by that neighbor node through TC messages.

3.2 Sleep mode extension

Sleep mode extension to basic OLSR protocol enhances its functionality by allowing the nodes to conserve their battery power. A node may wish to enter into sleep mode for a certain interval when it's battery level becomes low. Before going inactive, a node assures that it is not required for routing traffic of any other node. Moreover, the node's presence should be announced in the network through TC messages of some nodes. The nodes announcing its presence also buffer its incoming traffic. OLSR uses hop-by-hop routing which helps to bypass a sleeping node for all the traffic not destined to it. The packets are routed toward their destination according to the most recent information available at the neighbor nodes.

We define three states for a node: active, pre-sleep and sleep. A node is said to be active when it is participating in the network. A node is said to be a pre-sleep node when it wants to enter the power conservation mode (but has not yet entered into it). A node is said to be in "sleep" when it switches off its network interface or make it enter into a low power state.

3.3 Battery level thresholds

A criteria is required by a node in order to make decisions about when it should go to sleep and whether it can accept a request from a pre-sleep node to store its packets. In our algorithm, the battery level is considered as the only criteria. Battery power decreases with time and with traffic received and transmitted by a node. We define two threshold values: upper battery threshold and lower battery threshold. A node whose battery level gets lower than the upper battery threshold tries to enter into the sleep mode (pre-sleep state). However, it may not succeed in doing so in its negotiations, and its battery level continues to drop. A node whose battery level goes below the lower battery threshold is supposed to enter the sleep mode, even if the negotiations are unsuccessful.

Another threshold is defined as the sufficient energy level. A node can accept a request from a pre-sleep node to store the packets if its battery level is above the sufficient energy level. This level must be greater than the upper battery threshold because a node who intends to go itself into sleep mode should not store packets for other sleeping nodes.

4 Sleep mode operation

Sleep mode in OLSR works as follows: when a node runs short of battery power, it plans to go to sleep and its state becomes "pre-sleep". It stops transmitting periodic control messages: Hello messages so that none of its neighbors select it as an MPR anymore; and TC messages so that it is no more used in route calculations. It negotiates with its MPR neighbors if they can store its incoming traffic for the sleep mode duration. If some of its MPRs agree to do so, it can go to sleep for an agreed-upon sleep duration. When it wakes-up, it requests those MPRs to forward its stored packets, if any.

The MPRs storing the sleep node's packets also keep it in their *MPR Selector table* for the sleep mode duration. The MPR node thus announces the sleep node's address in its TC messages through which a sleeping node is known in the network. A sleeping node is not considered disconnected from the network and its packets are routed to its MPR nodes which buffer them during sleep period.

The message exchanged between the nodes for negotiating sleep mode operation is called Power Conservation (PC) message. PC message is broadcast to one hop neighbors only. A PC message contains a type field, a sleep period field to indicate the intended or accepted sleep duration, and an address field which contains a list of addresses depending upon the type of the PC message.

4.1 PC Request message to announce pre-sleep state

A pre-sleep node initiates the negotiations by sending a PC Request message which is broadcast to one hop neighbors only. The message contains the type as REQUEST1, the intended duration of the sleep mode and the list of addresses of pre-sleep node's MPRs. Through this message, the pre-sleep node requests its MPRs to store its incoming packets while it is in sleep mode.

Any node which receives a PC Request message removes the pre-sleep node from its neighbor table, and hence from the MPR set. The receiver then recalculates its MPR set and may reply with a PC Ack message, if requested (explained in Section 4.4). Moreover, the pre-sleep node is also removed from any route calculations.

If a pre-sleep node does not receive a reply of its PC Request message, it may time out and resend the request. After the first PC Request message, all subsequent retries will have the message type as REQUEST2. It indicates an increased demand for a power supporter MPR node, and it is more likely to be accepted (explained in Section 4.2).

4.2 PC Reply message to accept data buffering

If a receiver of PC Request message is an MPR of pre-sleep node, it has to decide if it can store the packets for the sender during the intended sleep period. If its battery level is greater than the sufficient energy level, it should accept to store the packets for pre-sleep node. However, if its battery level is lower than sufficient energy level but higher than the upper battery threshold, it should accept to buffer the traffic of pre-sleep node in case the PC message type is REQUEST2.

If a node agrees to keep the sleep node's packets, it sends a PC Reply message, in unicast, back to sender of the request. It then starts queuing the traffic that it receives for the sleep node. Moreover, it adds the sleep period to the life time of sleep node's entry in its *MPR Selector* table. In this way the MPR node keep announcing the sleep node's presence through its TC messages, during the sleep period.

When a pre-sleep node receives a PC Reply message, it can enter into sleep mode if its *MPR selector* set is empty and keeps only those MPRs in its neighbor table from which a PC Reply message is received.

4.3 PC Forward message to retrieve stored packets

When the sleep node wakes up, it resumes transmitting periodic Hello and TC messages to announce its presence. It then sends a PC Forward message to its MPRs which agreed to store its incoming traffic. A node which receives a PC Forward message sends back the stored packets of the sender of PC Forward message, if any.

If a node stores packets for a sleep node but does not receive a PC Forward message at the expiry of sleep mode duration, it checks to see if the sleep node is now active. If the node's entry is found in its neighbor table and a route exists to it then its stored packets are forwarded to it. Otherwise, the stored packets are discarded.

4.4 PC Ack message to accelerate negotiations

A pre-sleep node is assumed to behave as an MPR of some neighbor node until its *MPR Selector* table is not empty. If an acknowledgment may be sent when an *MPR Selector* removes the pre-sleep node from its MPR set, it will help in removing the entries from pre-sleep node's *MPR Selector* table without waiting for their normal expiry. Therefore, a pre-sleep node may request an acknowledgment from its *MPR Selectors*, in its PC Request message. In this way, it may enter into sleep mode more quickly. If the pre-sleep node is not requesting acknowledgments from its *MPR Selectors*, it may decide to go into sleep mode without waiting for MPR selector set to be empty.

4.5 PC Caution message to avoid network partitioning

Sleep mode operation of a node may cause the network to partition if that node were the only link between two partitions. It is desirable to maintain network connectivity and avoids partitions, if possible, created due to sleep nodes. PC Caution message is used as a warning message for this purpose, which indicates that a node is unable to reach some nodes of the network, via an alternate route if the requesting (pre-sleep) node goes to sleep. When a PC Request message is received, an MPR selector of the pre-sleep node checks the reachability with all nodes. After removing the pre-sleep node from its neighbor table, if it cannot find a route to any node, it sends a PC Caution message back to the pre-sleep node.

In order to prevent network partitioning, when a pre-sleep node receives a PC Caution message, it should not go into sleep mode if its battery level is greater than the lower battery threshold. If a pre-sleep node is connecting two partitions, it may keep on getting PC Caution messages each time it tries to enter the sleep mode. But eventually its battery level drops below the lower battery threshold, in which case it has to enter the sleep mode regardless of PC Caution messages.

5 Simulations

5.1 Simulation model

Our simulation model consisted of a 1500m x 1500m square network region containing about 50 to 200 randomly placed nodes for different topologies. In order to assure the existence of a path from a node to all other nodes in the network, we consider the connected network topologies only. We also assume that each link is a bi-directional link. For medium access, we used blocking of transmissions up to 2-hops, i.e., a node refrain from transmission if any of its neighbors up to 2-hops is transmitting. It was used to eliminate the hidden node problem. Mobility of nodes is not simulated in our study.

We used MoWiNet version 0.2b simulator in which the radio interface is modeled close to IEEE 802.11 MAC (without RTS/CTS) for a 2Mb/s channel with radio range 150m. In our simulations, data packets of 1KB average size were generated at 80 packets per second rate with randomly selected source and unicast destinations. These packets do not require an explicit acknowledgment because we assumed that acknowledgment is included in the packet transmission. The simulation starts with nodes having a random battery level between 50% and 100%. Each node consumes 0.01% of its battery power per second in idle mode. While transmitting or receiving a packet, it consumes additional 0.005% of its battery power per packet. No power is consumed in the sleep mode whose duration is 5 seconds.

5.2 Simulation results

An important concern for a power conservation scheme is the amount of buffer space required at the nodes which store packets for sleeping nodes. If traffic load is high and sleep duration is long, nodes have to have large buffer space to store frequently coming packets for a sleep node for longer durations. In our simulations, with a 40 node network and data packet arrival rate of 80 packets per second, the traffic of an active node comes out to be around 3 packets per second. More buffer space is required if a node happens to store packets for multiple sleep nodes at a time. The simulation results of Figure 1 show the maximum queue length as 17 and the average length as 5 packets. When the number of nodes increases, the share of data arrival rate per node decreases (keeping the overall network load constant). Therefore, in a large network, the maximum queue length is expected to be small accordingly.

Another important performance parameter is the end-to-end delay of the data packets. When a node is sleeping, its packets are buffered at its MPRs waiting for the node to wake up. This wait time which is added to the end-to-

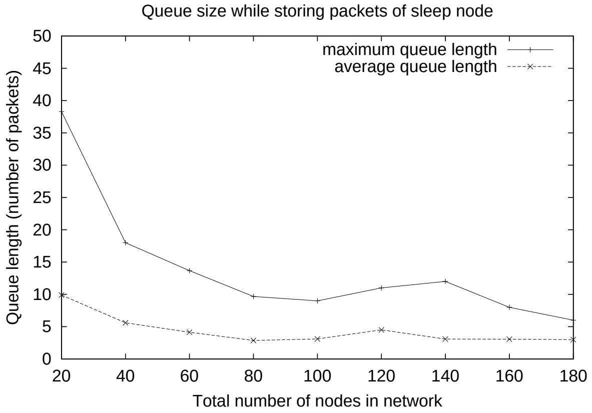

Figure 1. Queue length to store packets of sleep node

end delay of packets is proportional to the sleep duration, which was 5 seconds in our simulations. There is little or no effect on end-to-end delay of packets of other destinations due to sleep mode if the network remains connected. Figure 2 shows that the average end-to-end delay was around 3 seconds for a 40 node network while it was around 4.5 seconds for a large network.

The objective of a power conservation scheme is to enable battery operated nodes in a network to live for longer time. In a network with no power conservation mechanism, battery of the nodes is drained continuously resulting in earlier node deaths due to complete battery drainage. When a power conservation mechanism is employed, nodes cooperate in keeping the network functional by helping some low battery level nodes to save their resources. This results in longer life of individual nodes and hence a longer lifetime of the network overall. Figure 3 shows that without power conservation mechanism, the network lifetime ends (first node goes to death) much earlier, i.e. 340 seconds after the start of simulation, as compared to 510 seconds in case of a network with power conservation. Moreover, the rate of nodes going dead is much slower in a network using power conservation scheme.

6 Conclusions

Results of the simulation show that a network without power conservation mechanism consumes the battery power of nodes continuously resulting in a

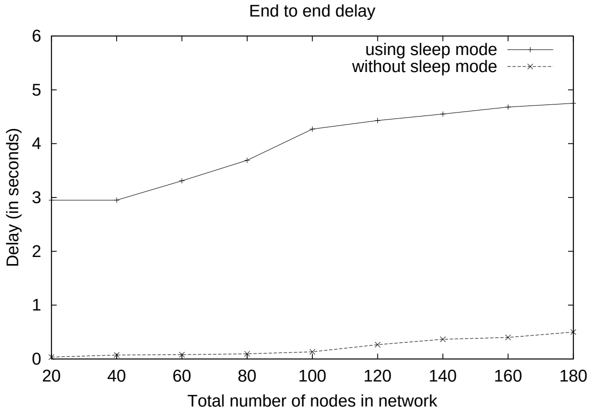

Figure 2. End to end delay for packets of sleeping node

faster rate of node deaths having their batteries completely drained off. On the other hand, a sleep mode gives a small rest intervals to a node resulting in a longer life time of network. The amount of buffer space required to store packets of a sleep node depends on load on the network. Our results show that the required buffer length was quite small for a moderately loaded network.

Sleep mode however incur longer end-to-end delays for packets of sleeping nodes. This increase in end-to-end delay is proportional to sleep duration. If an application is delay sensitive or requires an acknowledgment in a specified time (e.g. TCP), the sleep duration of a node must be tuned accordingly. In that case, the nodes may go into sleep mode more frequently, but for shorter durations hence adapting to delay sensitive applications.

References

1. R. Kravets and P. Krishnan. Power Management Techniques for Mobile Communication. In *Proc. Mobicom'98*, October 1998.
2. Laura Marie Feeney. An Energy-consumption Model for Performance Analysis of Routing Protocols for Mobile Adhoc Networks In *Proc. of 45th IETF Meeting, MANET Working Group.*
3. Laura M. Feeney and Martin Nilsson. Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment In *Proc. of IEEE INFOCOM*, April 2001. Anchorage, AK.

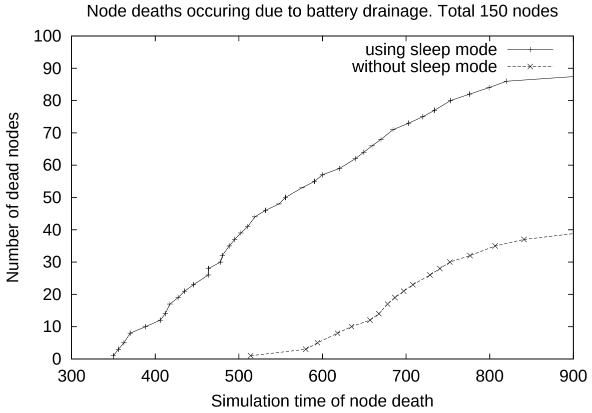

Figure 3. Network lifetime in terms of node deaths

4. David Hinds. Linux PCMCIA Guide, http://pcmcia.sourceforge.org.
5. ETSI STC-RES10 Committee. Discontinuous Transmission (dtx) for Enhanced Full Rate (EFR) Speech Traffic Channels (GSM 06.81 version 5.2.1 release 1996), 2nd ed., April 2000.
6. S. Singh and C.S Raghavendra. PAMAS: Power Aware Multi Access Protocol with Signalling for Ad Hoc Networks. *Computer Communications Review*, 28(3):5–26, 1998.
7. Ya Xu, John Heidemann, and Deborah Estrin. Adaptive Energy-Conserving Routing for Multihop Adhoc Networks. Technical Report 527, USC/ISI, October 2000.
8. Ben Lee, Hee Yong Youn, and Chansu Yu et al. Non-Blocking, Localized Routing Algorithm for Balanced Energy Consumption in Mobile Adhoc Networks In *Proc. of MASCOT'2001*, August 2001. Cincinnati, OH.
9. P. Jacquet, P. Mühlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized Link State Routing Protocol for Ad Hoc Networks. In *Proc. IEEE INMIC 2001*, December 2001. Lahore, Pakistan.
10. A. Qayyum, L. Viennot, and A. Laouiti. Multipoint Relaying for Flooding Broadcast Messages in Mobile Wireless Networks, In *Proc. HICSS-35* Big Island, Hawaii, USA, 7-10 January, 2002
11. A. Qayyum. *Analysis and Evaluation of Channel Access Schemes and Routing Protocols in Wireless LANs.* PhD thesis, Université de Paris-sud, Orsay, France, 2000.