PIES: Protocol Independent Energy Saving Technique for Mobile Ad Hoc Networks

By

Yasser S. Gadallah

A Thesis submitted to

The Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Ottawa-Carleton Institute for Electrical Engineering

Faculty of Engineering

Department of Systems and Computer Engineering

Carleton University

Ottawa, Ontario, Canada

April 22, 2005

© 2005 Yasser S. Gadallah

Abstract

Mobile communications have become very popular nowadays, especially with the advent of light-weight portable devices that made it convenient to perform many types of tasks while on the move. This has led to rapid advances in the mobile ad hoc network (MANET) field of research. Ad hoc networks are composed of mobile wireless nodes that are generally moving freely within a certain area. These nodes cooperate together in order to route traffic from a source to a specific destination. Several routing protocols have been devised to handle the routing duties within ad hoc networks.

Battery energy is the most scarce resource on which the continued functionality of a mobile ad hoc network depends. Several schemes have been developed to address the issue of energy efficiency with varying degrees of success. In this thesis, we present a new energy-efficient algorithm that we call PIES, which stands for *Protocol-Independent Energy Saving*. The PIES algorithm presents a new energy saving technique that helps conserve energy that is consumed by the wireless interfaces of the network nodes. PIES is not a routing algorithm, rather it is an algorithm that works in conjunction with existing routing protocols and helps those protocols make decisions regarding energy conservation. PIES functionality conforms to the principle of energy fairness amongst network nodes and does not intervene with the core functionality of the routing protocol. It is a distributed algorithm whose functionality does not depend on any single node or set of nodes within the network. In addition, PIES does not introduce any significant additional traffic or energy costs to the network and its nodes and can be configured to have no additional traffic or energy costs. Our experimental results show that PIES introduces considerable energy savings and works equally well with various categories of routing protocols. This is done in a fair manner to all nodes within the network. Our experiments also show that PIES functionality is consistent with the increase of network traffic and population and increases network lifetime by about 70%.

In addition to the simulation studies, we analyzed the impact of PIES major parameters on protocol performance and validated them through simulation. This provides guidance to network operators in setting these parameters.

Acknowledgements

I would like to thank my research supervisor, Professor Thomas Kunz, for his guidance and insistence on achieving high quality research. The numerous fruitful discussions that we have had during the course of this study have made a significant difference in the direction and conclusion of this work.

I would also like to express my deep gratitude to my wife and daughters for their patience, sacrifice and constant support and encouragement during the years I have spent working on my Ph.D. studies.

Table of Content

List of Tables

List of Figures

List of Acronyms

AODV	Ad hoc On-demand Distance Victor routing
ATIM	Ad hoc Traffic Indication Map
CBR	Constant Bit Rate
CMDR	Conditional Minimum Drain Rate
CMMBCR	Conditional Max-Min Battery Cost Routing
CTS	Clear To Send
DIFS	Distributed Interframe Space
DSDV	Destination-Sequenced Distance-Victor routing
DSR	Dynamic Source Routing
DPC	Distributed Power Control
FA	Flow Augmentation
FR	Flow Redirection
GAF	Geographic Adaptive Fidelity
LAN	Local Area Network
MAC	Medium Access Control
MACA	Multiple Access with Collision Avoidance
MANET	Mobile Ad Hoc Networks
MDR	Minimum Drain Rate
MMBCR	Max-Min Battery Cost Routing
MPR	Multipoint Relay
MTRP	Minimum total Transmission Power

ns2	Network Simulator version 2
OLSR	Optimized Link State Routing
PAMAS	Power Aware Multi-Access with Signaling
PDR	Packet Delivery Ratio
PIES	Protocols Independent Energy Saving
QoS	Quality of Service
RREQ	Route Request
RTS	Ready To Send
SIFS	Short Interframe Space
TCP	Transmission Control Protocol
VBS	Virtual Base Stations

Chapter 1: Introduction

This research presents a solution to some of the energy efficiency issues in connection with communications in Mobile Ad hoc Networks (MANET). In this chapter, we give a brief introduction to mobile ad hoc networks and the elements that affect their operation. We limit our attention to those MANET aspects that are directly linked to the research covered by this study. We then describe our research motivations and goals and present a list of thesis contributions. Finally, we describe the organization of this thesis.

1.1 Mobile Ad Hoc Networks: Characteristics and Applications

Recently, mobile communications have flourished extensively with the technological advances made in areas of new portable devices and wireless communications. This has led to many advances in the area of mobile ad hoc networking. A mobile ad hoc network consists of mobile devices communicating with each other via wireless connections to both exchange information of mutual interest as well as to maintain the network connectivity in general. These devices are generally free to move about arbitrarily, and could be located on airplanes, in cars, with people, etc. We refer to these mobile devices as "*network nodes*" within this study. Therefore, an ad hoc network is generally considered an infrastructureless network that relies on its nodes to maintain its topology. This implies that the different nodes are expected to perform, in addition to their normal function e.g. as a computing device, the routing function that is normally done by routers within the Internet infrastructure networks. An example of an ad hoc network is illustrated in Figure 1-1. The figure shows a network that consists of seven mobile nodes. The wireless links between the nodes that lie within range of each other are illustrated by bi-directional arrows. In this network, if node 1 needs to communicate with node 2, for example, it will need to do so via node 4, as one of the possible communication paths. If a node is within range of another node, for example nodes 4 and 7, they can communicate with each other directly. But, if node 7 decides to move out of range from node 4, this direct communication would no longer be possible and the two nodes would have to continue their communication via one or more other nodes.

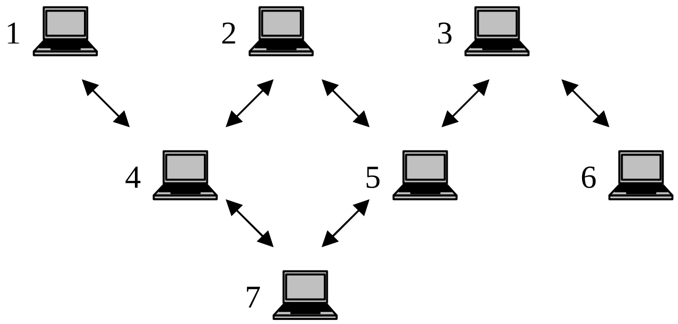


Figure 1-1: A mobile ad hoc Network

In this thesis, we focus on ad hoc networks that conform to the following characteristics [8], [32]:

- The network nodes are using IP, the Internet Protocol
- The nodes could be far apart in such a way that not all of them are within range of each other
- The nodes are generally mobile and therefore two nodes that are within range from each other at some point of time may be out of range later

- The nodes are able to contribute to the topology maintenance operations of the network
- The network is generally bandwidth constrained as wireless links generally have lower capacity than wired links. This is a large factor of consideration in the design of the protocols that are used for the operation of these networks
- The nodes generally possess a limited amount of battery energy

There are numerous applications of the MANET technology. Here are some examples of their use [32]:

- *Conferencing*: where a network infrastructure is missing while conference or meeting attendees still require exchanging emails and information regarding a certain project or task at hand.
- *Emergency Services*: for search and rescue missions, for example, where several emergency workers need to exchange information over a vast searching area with no existent network infrastructure.
- *Military Operations*: similar to the previous case, a group of military personnel may require exchanging operational information while in the field with no access to a friendly network infrastructure.
- *Sensor Networks*: in which a set of sensors with wireless transceivers can be randomly spread over an area for which terrain or environmental information gathering is needed. These sensors can cooperate in collecting and assembling this information for analysis purposes.
- *Wireless Mesh Networks*: A mesh network allows nodes or access points to communicate with other nodes without being routed through a central switch

point, eliminating centralized failure, and providing self-healing and self-organization. Intelligence is distributed from switches to access points by incorporating a grid-like topology. Network nodes act as routers. This type of networks can be used where wired LANs cannot be established easily, or where cost of establishing them is too high.

1.2 Factors Affecting the Operation of MANETs

There are many factors that affect the operation of a mobile ad hoc network. These factors have to be considered while designing any solution or protocol that addresses MANET related issues. In this section the main factors are presented together with their effect on the network operation.

1.2.1 Wireless Communication

Ad hoc networks exhibit reduced data rates as compared to wired networks, with an order of magnitude difference between ad hoc and wired networks. Therefore, this limited bandwidth should be an important factor to consider in the design of any ad hoc network protocol. It is important to reduce packet transmission overheads (e.g. protocol-specific control packets) between network nodes. In addition to this bandwidth limitation, wireless media also experience higher error rates in comparison to wired networks. This presents a problem especially with protocols such as TCP which were designed to interpret a lost packet as an indication of network congestion. This, in case of ad hoc networks, causes an even larger performance issue to users when transient noise, for example, causes a temporary increase in errors.

1.2.2 Scalability

Node mobility in ad hoc networks introduces scalability issues due to the fact that routing changes as nodes move. This requires control messages to be sent around the network in order to update connectivity information. The quicker the movements of nodes relative to each other, the more often the control messages have to be sent to maintain network topology information. This imposes additional load on network bandwidth which is already limited, as explained in the previous point. Therefore, the number of control messages required by a protocol intended for ad hoc networks should be an important design factor. Evaluation of such algorithms should include their ability to scale with the growth of network population, traffic and node mobility. From this point of view, the lower the control traffic of a protocol, the more accepted it is for operation within wireless ad hoc networks.

1.2.3 Energy Limitations

Generally speaking, nodes within ad hoc networks rely on limited energy sources, usually batteries, for their operation. While energy is consumed by different aspects of the functionality of a mobile node, we focus our attention in this study on those aspects that relate to communication between network nodes. Nodes consume energy when they transmit data to a desired destination, when they forward data while acting as intermediate nodes between source and destination nodes, or when they listen to a channel. Since nodes cannot operate without energy, and since energy in mobile nodes could be highly limited due to the generally limited battery power, this can be considered one of the most important limiting factors in operating an ad hoc network. Every time a node transmits, receives or listens to a communication medium, it consumes energy as will be explained in detail later in this thesis. This underscores the importance of energy conservation in connection to communications in ad hoc networks.

This research focuses on creating an energy efficient technique that works in conjunction with existing ad hoc routing protocols.

1.2.4 Routing Techniques

Generally, nodes within an ad hoc network perform the routing duties that are handled by fixed routers in wired networks. Therefore, nodes that are willing to contribute to routing duties within the network perform them in addition to their original functions. These nodes would therefore have to run a certain routing protocol in order to be able to contribute to the routing function. Several routing protocols have been created for ad hoc networks; each of which possesses its own characteristics that differentiates it from the others. The routing technique is an important factor that affects the performance of the ad hoc network. This is because the strategy of the routing protocol determines such important aspects of the network operation as latency, congestion, power consumption, etc.

Routing protocols can be divided into two categories [38]: table-driven (or proactive) and source-initiated on-demand driven (or reactive) protocols. The table-driven protocols' strategy is to maintain up-to-date routing information in routing tables that are maintained at each network node. These protocols respond to changes in network topology by propagating updates throughout the network to maintain a consistent network view. Example protocols are the DSDV routing protocol [33] and the OLSR routing protocol [21]. Source-initiated on-demand routing protocols on the other hand create routes only when desired by the source node. When a source node needs to communicate with a destination node, it initiates a route discovery process within the network. The process is completed when at least one route is discovered. Once a route is established, it is maintained by some sort of maintenance procedure until the route is no longer needed or the destination is not at all accessible. Example protocols are the DSR [24] and the AODV [34] routing protocols.

1.3 Performance Metrics

Several algorithms have been developed in support of ad hoc networks' operation. Some of these algorithms are purely routing algorithms, and some of them are used in conjunction with routing algorithms for other complementary functions such as energy conservation. In order to evaluate these algorithms and to test their performance relative to each other, several metrics have been widely used [32], [35]. Examples of these metrics are as follows:

- *Average packet delivery ratio*: ratio of the data packets received at the destination nodes to the packets that were sent by the sources.
- *Average packet latency*: includes all the delays encountered by the packet at the different hops from the time it was sent by the source until the time it was received at the destination.
- *Route acquisition time*: the average time period between requesting a route and acquiring it. This metric is of particular concern to routing protocols that belong to the "on-demand" category of routing algorithms.
- *Routing load*: number of routing packets (and supporting protocol control packets) transmitted per data packet delivered at the destination.

As far as network context is concerned, the following parameters are also essential in the evaluation and comparison of ad hoc network algorithms:

- *Network size*: Measured in the number of nodes.
- *Network area dimensions*: This also includes the shape of the area
- *Rate of Change of network topology*: This is reflected by the speed of nodes as well as pause time between movements
- *Traffic load*: This includes the following parameters:
    - Traffic type (e.g. CBR versus TCP)
    - Data rate in packets per second
    - Packet size in bytes. This parameter combined with the data rate is sometimes referred to as the throughput and is usually measured in Kbps
    - Number of source/destination pairs
- *Node energy parameters*: This depends on whether an energy efficient algorithm is in use and generally includes parameters such as sleep times, frequency of the sleep state, number of sleeping nodes at a time, etc.

Throughout this study, we will be using some of these metrics and parameters as well as other energy-efficiency related metrics that will be discussed in the next chapter to evaluate our approach.

1.4 Motivations and Goals of this Research

Energy is the most scarce resource for the operation of the mobile ad hoc networks. As we will see in Chapter 2, idle energy consumption is responsible for a large portion of the overall energy consumption in the wireless interfaces of the mobile nodes. Therefore, it is crucial to energy conservation efforts that this source of energy is eliminated or reduced.

Our goal in this research is to create a new energy conservation scheme that works on reducing idle energy consumption. This scheme works with existing routing algorithms of all categories. It aims at achieving energy conservation in a manner fair to all network nodes. It is distributed in nature, and its functionality is independent of the strategy and architecture of the routing protocol. We will discuss our research objectives in more detail in Chapter 2.

1.5 Contributions of this Thesis

The main contribution of this thesis is investigating energy-efficiency issues that relate to communication in mobile ad hoc networks. Mobile nodes generally possess limited sources of energy on which the health of these nodes and the network is mainly dependent. Therefore, conserving this energy is crucial for the extended functionality of such networks, and hence this is the target of our research.

The contribution of this research can be summarized in the following points:

1. Studying energy-efficiency issues associated with communication between nodes that constitute a mobile ad hoc network. This includes a case study [16] that illustrates these issues and highlights the existence of an opportunity to address them (Chapter 2).
2. Classification and evaluation of energy-conserving techniques in mobile ad hoc networks (Chapter 3).
3. Design of a novel energy-conservation technique [17] that works in conjunction with existing routing protocols. This technique is fair, modular in nature, distributed (does not depend on a single node or set of nodes for its functionality)

and its functionality is independent of the nature and semantics of the routing protocol (Chapter 4).

4. Implementation of the new algorithm in the ns2 simulator [13] as well as integrating it with the AODV, OLSR and DSR routing algorithms (Chapter 5).
5. Full evaluation of the performance of the new technique which includes its performance with different types of routing protocols, scalability, effect on network lifetime, its performance with different traffic types, the effect of changing its parameters on performance and a comparison with another energy-conserving algorithm of the same category (Chapter 5).
6. Analytical study of the effect of the new technique on latency [18] and a comparison with experimental results (Chapter 6).
7. Analytical study of the effect of the algorithm on network throughput and a comparison with experimental results (Chapter 6).

This research has resulted in the following publications so far:

- Y. Gadallah and T. Kunz, “A Protocol Independent Energy Saving Technique for Mobile Ad Hoc Networks”, accepted for the special issue on Mobile and Wireless Networking, International Journal of High Performance Computing and Networking (IJHPCN). Expected to be published early 2005.
- Y. Gadallah and T. Kunz, "PIES: Protocol independent energy saving algorithm", Proceedings of the 2004 Workshop on Mobile and Wireless Networking, Montreal, Canada, pages 4-12, August 2004, IEEE Computer Society Press 2004, ISBN 0-7695-2198-3.

- Y. Gadallah and T. Kunz, "Energy consumption in ad-hoc routing protocols: Comparing DSR, AODV and TORA", Proceedings of the 1st International Conference on Ad-Hoc Networks and Wireless, Toronto, Canada, pages 161-176, September 2002.

1.6 Organization of this Thesis

This thesis is organized as follows. This introductory chapter briefly describes the ad hoc network concepts and entities that are used throughout this work. In Chapter 2, some ad hoc networking environment energy-related background is given and the issues associated with energy conservation are discussed in detail. This includes a description of the network operation from an energy point of view, energy models and energy metrics. It also includes a case study that paves the way to the description of the problem that this work addresses. It concludes with a description of our research objectives. Chapter 3 includes a description of related work. It includes a discussion of some of the most important techniques that have been published to address energy-efficiency issues. Chapter 4 includes a description of our solution to address the energy-efficiency problems that we presented in Chapter 2. Chapter 5 includes detailed evaluation experiments of our solution. Chapter 6 presents an analytical discussion of the algorithm’s dominant parameters as well as their effect on latency and network throughput. Chapter 7 concludes the study and suggests areas for future research.

Chapter 2: Energy Efficiency Issues in Mobile Ad Hoc Networks

2.1 Introduction

As we mentioned in the previous chapter, nodes within an ad hoc network generally rely on batteries (or exhaustive energy sources) for power. Since these energy sources have a limited lifetime, power availability is one of the most important constraints for the operation of the ad hoc network. There are different sources of power consumption in a mobile node. Communication is one of the main sources of energy consumption [43]. Since the rate of battery performance improvement is rather slow currently, and in the absence of breakthroughs in this field, other measures have to be taken to achieve the goal of getting more performance out of the currently available battery resources. Within this study, we focus our efforts on methods to reduce the power consumed in communications between ad hoc network nodes.

Communications in ad hoc networks are done via using the RF transceiver at the source, intermediate and destination nodes to exchange information. The source node sends control and data messages which are received by one or more receiving nodes, depending on the message type. The receiving node could be the intended receiver of a packet, or it could be on the path to the end destination (when the destination is not within range from the source) and in this case it acts as a forwarder of the traffic. In order to address the energy efficiency issues in the communications within ad hoc networks, it is important to understand the energy model which represents the power consumption behavior in the ad hoc network node wireless interfaces.

2.2 Energy Models

Several studies have dealt with measuring energy consumption in the wireless interfaces of mobile nodes, e.g. [14][15],[28],[43] to determine the exact sources of energy consumption in the wireless interfaces. These studies examined the different modes of operation of the wireless interfaces. It was found that a mobile node’s wireless interface consumes energy not only while communicating with other nodes, but also while in idle mode, i.e. when the node is listening to the medium but not handling packets. Following are the types of energy consumption that have been identified:

- Energy consumed while sending a packet
- Energy consumed while receiving a packet
- Energy consumed while in idle mode
- Energy consumed while in sleep mode which occurs when the wireless interface of the mobile node is turned off

It should be noted that the energy consumed during sending a packet is the largest source of energy consumption of all modes. This is followed by the energy consumption during receiving a packet. Despite the fact that while in idle mode the node does not actually handle data communication operations, it was found that the wireless interface consumes a considerable amount of energy nevertheless. This amount approaches the amount that is consumed in the receive operation. Later in this chapter we present the results of some simulation studies that we have performed to determine the ratio of the average energy consumed in idle mode relative to the overall energy consumption. Idle energy is a wasted energy that should be eliminated or reduced through energy-efficient schemes. Through energy consumption measurements studies, experiments have also been conducted to determine the power consumption patterns in the different active modes. In some experiments, the instantaneous power consumption per communication mode, e.g. send, receive, idle and sleep modes, has been measured. Some experiments went even further to include more details about the energy consumption pattern per subtype of the operation [14],[15]. For example, the cases of unicast and broadcast are considered to have different costs. This has been explained based on the fact that unicast operations in IEEE 802.11 involve the exchange of control packets between the sending and receiving nodes while broadcast operations do not involve such an exchange. However, these studies did not directly address cases of repeated resending of control packets that may happen due to glitches in the transmission operations over the wireless communication channels. It has been shown [39] that energy consumed in the retransmit operations is responsible for a considerable amount of energy consumption. Since this case cannot be avoided with the use of energy-efficient algorithms, especially in the transition between node wakeup and sleep times, using the model described by [14],[15] may introduce some inaccuracies.

Throughout this study, we have benefited from [14] and other measurements studies in the usage of the values that they measured for the energy consumption of the wireless interfaces during the different modes.

2.3 Energy Consumption Overhead

As indicated above, energy consumption through communications in MANET is caused by several sources. Some of these sources are useful while others are considered a waste that should be eliminated or reduced in future designs for energy efficient schemes. Some of the main sources of wasted energy in wireless interfaces can be attributed to the following [20]:

- *Idle condition*: With interfaces sitting idle most of the time especially in applications such as e-mail or web-browsing [43].
- *Collisions*: In these situations, which happen mainly at high load conditions, data involved in the collision become useless and the energy used to communicate this data is lost.
- *Protocol overhead*: This refers to protocol-specific control messages which impose additional energy requirements on top of what is necessary to transmit payload traffic. Managing this source adds a design requirement to MANET targeted protocols.
- *High error rate*: This is typical for wireless links. This source of energy consumption overhead actually has three components to it:
    - Data involved in erroneous transmissions becomes invalid and hence energy consumed in their communication is wasted.
    - Energy is used for error control mechanisms such as those used at the data link level to reduce the impact of errors.

    - Some routing protocols may interpret delays due to temporary error conditions as routes becoming invalid. This leads to flooding the network with route requests to substitute for existing valid routes that are thought to have become broken. This could be a large source of wasted energy.

Energy efficient schemes are needed to address the first issue. Possible solutions would be to put nodes to sleep for periods of time in order to avoid wasting energy in idle conditions. In this case, several questions would need to be answered by such algorithms, for example:

- When a node is put to sleep?
- When does it wake up?
- How would other nodes know if a certain node is asleep?

The second issue may be unavoidable with heavy load conditions and in high mobility conditions and is best handled by MAC level back-off strategies such as the one available with the IEEE 802.11 MAC [50]. The third issue should be handled through a conservative design strategy of ad hoc network protocols by restricting control messages to the narrowest possible limits. The fourth issue can be handled at different layers; in particular the MAC and network layers. At the MAC layer, a possible solution would be to increase the transmission power upon the detection of error conditions so that the SNR would increase in the hope of overcoming the erroneous conditions. At the network layer, the routing protocol should have enhanced fault tolerance in order to handle such error conditions. However, this should be balanced with the need to detect “real” fault conditions so that unnecessarily long delays can be avoided.

2.4 Energy Efficiency Metrics

Energy efficiency metrics are needed to both devise and evaluate energy conservation schemes. Throughout this work, we assume the following:

- Only unicast routing is considered (i.e. multicast is not covered in this study).
- A link layer acknowledgements mechanism (such as that provided by an IEEE 802.11 wireless LAN) is available.
- Only energy consumed during node communications will be focused on.

Some studies, e.g. [42], discuss metrics for energy efficiency in ad hoc networks. Several distinct metrics can be used to compare the energy-efficiency of a routing protocol. The selection of appropriate metrics depends on the overall goal of the ad-hoc network deployment. Examples of the metrics that are used to manage energy consumption are:

- Minimize energy consumed/packet,
- Maximize time to network partition,
- Minimize variance in node energy levels,
- Minimize cost/packet, and
- Minimize maximum node cost.

For routing algorithms that find shortest paths, the first and fourth metrics are used. The “Minimize energy consumed/packet” metric basically addresses the average energy consumed per packet, over the number of hops that are traversed by this packet. Hence, if $e_{i,j}$ is the energy consumed in transmitting one packet between nodes i and j, then the energy consumed for packet x, e_x, transmitted from source to destination over n hops, is:

$$e_x = \sum_{i=1}^{n} e_{i,i+1} \tag{2-1}$$

As for the “Minimize cost/packet” metric, a cost function has to be defined. The total cost (or energy) of sending a packet along some path is the sum of node costs along that path. This metric is generally used if we are trying to derive an algorithm that maximizes the life of all nodes in the network. There is an important difference between this metric and the first one (Minimize energy consumed/packet). The former makes sure that the nodes that are low in energy do not lie on many paths while the latter does not consider the energy level of the nodes as long as the path a packet takes will result in minimum energy consumption for that packet.

If ensuring fair energy distribution amongst network nodes is of concern, then the “Minimize variance in node energy levels” metric must be used. This metric ensures that all nodes in the network remain up and running together for as long as possible. A way to achieve this could be via selecting the routing path (if several are available) based on the energy levels of the different nodes on the path. That is, one would determine the smallest value of node energies on a certain path, and compare it with the corresponding value on the other paths. The path with largest bottleneck energy value is selected. We will show later that, as one of the aspects of our energy conserving strategy, we have opted to using this method as one of our measures to ensure fairness amongst network nodes.

In the next section, however, since we are only comparing existing routing algorithms with regard to the energy consumption, we will be using the average overall energy consumed combined with the ratio of packet delivery for our comparison and conclusions. That is, we compare the algorithms with regard to average overall amount of energy used to route packets together with packet delivery performance. We also use the standard deviation of remaining node energies as a measure for energy distribution between network nodes, when comparing the algorithms.

2.5 An Example to Illustrate Energy Consumption Issues

To get an idea about the nature of some of the energy consumption issues that are encountered in ad hoc networks, we performed a comparison study of some popular ad hoc routing algorithms [16]. In this section, we present the results of comparing the DSR [24] and AODV [34] routing algorithms. This gives some insight into the issues we are trying to address in this research.

These two algorithms use the shortest-path routing strategy and do not have an energy conservation technique. We demonstrate the difference between the algorithms in terms of their energy consumption. We used the ns2 simulator [13] to conduct our investigation. We also used the CMU Monarch Project's [12] wireless and mobility enhancements to ns2.

We use a relatively high value for the maximum node speed and run simulations for different pause time values for this speed. Our goal is to examine and compare the energy consumption patterns at a mobility condition that would cause the topology to change relatively fast. Note that the only reasonable energy consumption comparison that can be performed is at the system level, i.e. average energy comparison. It does not make sense from the point of view of examining different routing strategies to perform the comparison at the individual node level, since the different algorithms may select different nodes to route packets. We did not perform detailed performance comparisons of the algorithms in terms of their ability to deliver packets under different conditions since this was the subject of other studies e.g. [6],[23],[35]. We only compared their energy consumption patterns under different conditions and the corresponding packet delivery ratios. This was to ensure that a lower energy consumption did not come at the expense of the algorithm's packet delivery performance. Tables 2-1 and 2-2 show the values that we used for the different simulations. Table 2-1 shows the common simulation parameters. The results of our simulations represent the average of 5 runs each with different mobility scenario. We used the same traffic pattern for all simulation experiments. Our movement scenarios are characterized by a pause time. Each node starts the simulation by remaining stationary for "pause time" seconds. It then moves to a selected random destination at a speed that is uniformly distributed between zero and the maximum speed that we selected for the simulation. This pattern then repeats itself over and over until the end of the simulation. We ran the simulations for different pause times for each of the mobility scenarios. The power consumption values for the different modes of operation are listed in Table 2-2.

Table 2-1: Simulation parameters for the case study

Number of nodes	50
Dimensions of simulation area (m×m)	1500×300
Initial node energy (Joules)	1000
Simulation time (seconds)	600
Traffic type	CBR, 3 packets/s
Packet size (bytes)	512
Number of traffic connections	20
Maximum speed (m/s)	20

Table 2-2: Energy model parameters for the case study

Rx Power Consumption	1.0 W
Tx Power Consumption	1.4 W
Idle Power Consumption	0.83 W

We selected CBR for traffic instead of TCP to be able to perform the comparison between the algorithms under equal conditions since TCP changes the load (the number of packets it sends) based on the network conditions, which would prevent a direct comparison between the algorithms. With these simulation conditions, we obtained results that are plotted in Figure 2-1. The results indicate that the energy consumption patterns of DSR and AODV are quite close, with DSR performing slightly better than AODV especially with high mobility conditions. As part of the energy performance, we also measured the standard deviation of remaining node energies in order to see how the

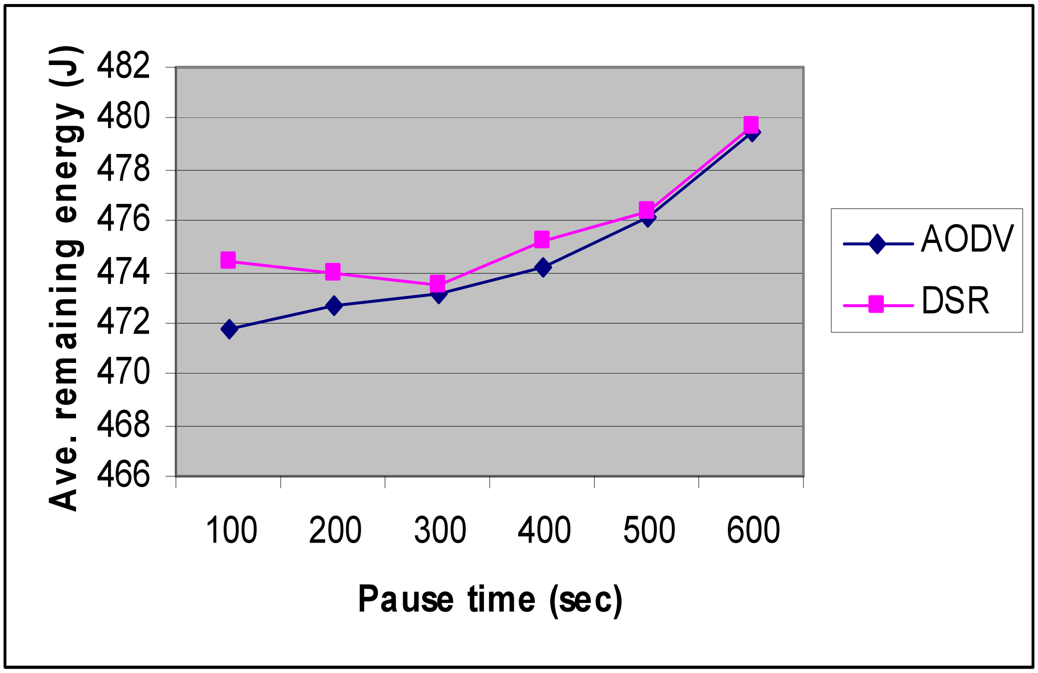


Figure 2-1: Energy consumption patterns for DSR and AODV

nodes are utilized in both AODV and DSR. It is worth mentioning that the lower the standard deviation, the more balanced node utilization is. Figure 2-2 shows these results. The figure also shows that DSR and AODV perform quite closely with AODV showing slightly better results than DSR especially at higher mobility. This is due to the fact that all nodes have to send AODV HELLO messages periodically. This implies more equal utilization of network nodes.

However, if we only consider these energy consumption measures, the picture is incomplete. Therefore, packet delivery ratios are important for a complete understanding of the results.

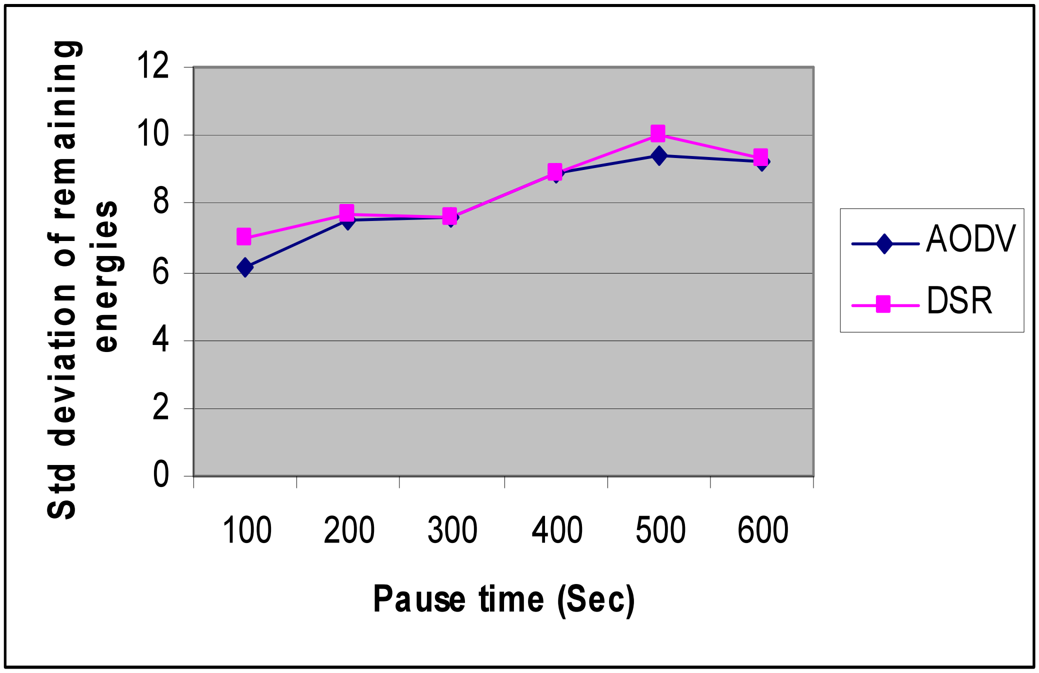


Figure 2-2: Node utilization under the AODV and DSR routing protocols

Figure 2-3 shows the corresponding performance of the algorithms in terms of the packet delivery ratios. From this chart, we see that the data delivery performance is almost identical for DSR and AODV, for the conditions of our simulation. From this we see that for the simulation cases that we have studied, the DSR algorithm offers the best combination of energy consumption performance and the data delivery performance. It is followed quite closely by the AODV algorithm to the extent that they both seem to be almost identical for most of our cases.

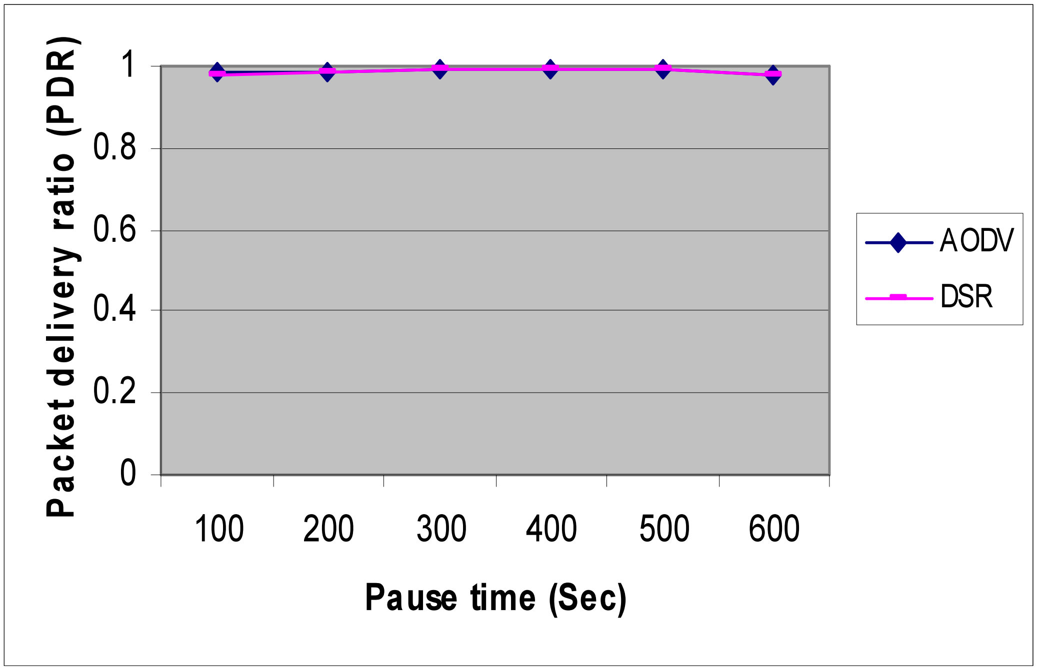


Figure 2-3: Packet delivery ratio for AODV and DSR

We also measured the two algorithms with respect to idle energy consumption, see Figure 2-4. In general, idle energy consumption is an obvious candidate for energy conservation efforts. We found that AODV and DSR are similar in this regard. The idle energy consumption constitutes well above half of the energy consumption for both algorithms. It is worth mentioning that the results concerning idle energy are somewhat different from those obtained through other studies e.g. [47]. The reason is the difference in the used energy model in these particular simulations. In any case, it is clear from most studies that idle energy accounts for at least 50% of total energy consumption in ad hoc network operation. So identifying when nodes can go from a relatively high-energy idle state to a low-energy sleep state is one important avenue for energy conservation. Also, despite the fact that idle energy provides an inherent source of energy balance between different nodes to some extent, we can still see some energy imbalance between network nodes. Energy fairness, therefore, is another area that needs to be focused on. This becomes of special importance as idle energy gets reduced due to some energy conservation strategy, as this undesired source of balance that exists with algorithms with no energy conservation strategies would be reduced or eliminated.

2.6 Objectives of this Research

As indicated in the discussion and results above, many routing algorithms have been created based on various strategies with no policy to address energy efficiency issues. We have seen some examples of these algorithms in which the nodes spend a large percentage of their energy in idle mode, which is considered a source of large energy waste. We have also seen that despite the existence of idle energy which, in a way, introduces a source of energy balance between network nodes, there still exists some imbalance between node energies.

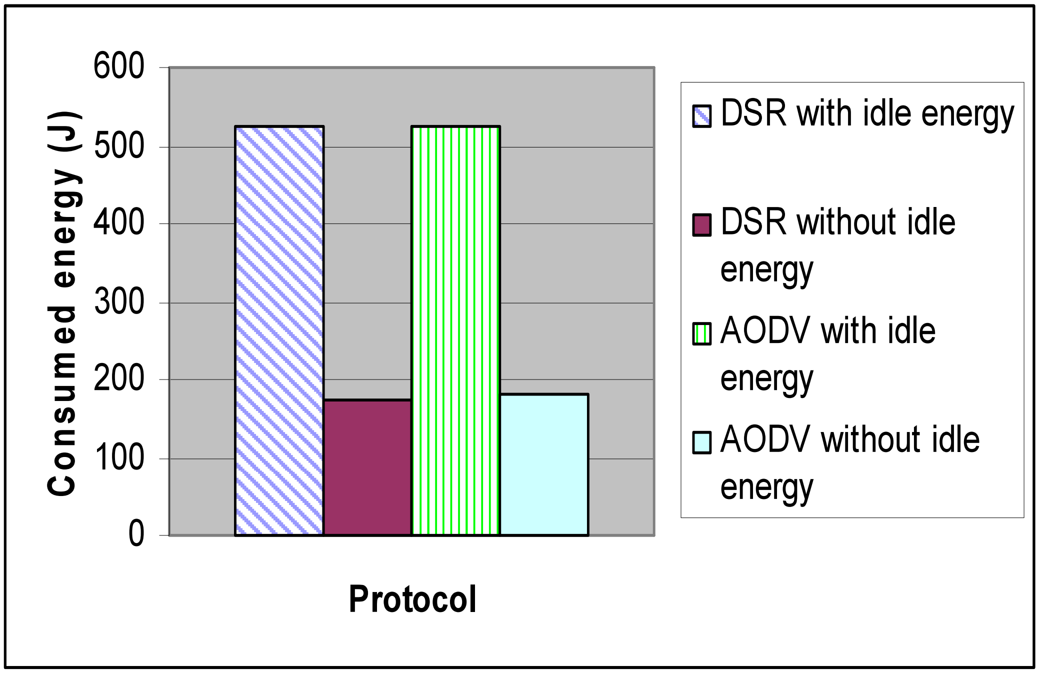


Figure 2-4: Energy consumption with and without idle energy in DSR and AODV (pause 400 sec)

In order to address these issues, a different strategy that takes energy efficiency into consideration needs to be followed. The possibilities for this strategy range from creating new energy-efficient routing protocols to enhancing existing ones to become energy efficient. There have been several studies that explored this issue and we will be discussing some of these studies in the next chapter. Emerging from the discussions so far are the following main issues that we would like to address within the scope of this work:

- A large amount of energy is wasted while the wireless interfaces of the mobile nodes are idle
- Energy imbalance between network nodes
- Existence of routing protocols with energy-inefficient characteristics

Considering these issues as well as our discussion around wireless ad hoc network issues, our goal in this research is to create an energy efficient algorithm that achieves the following main objectives:

- Fair energy conservation.
- Fully distributed operation: this algorithm does not depend on a particular node or set of network nodes for its functionality. This ensures a robust operation of the algorithm especially in case of node failures.
- Modular architecture: enables it to integrate relatively easily with existing routing protocols without the need to make custom modifications to it or major modifications to the routing algorithm upon integration.

Therefore, our focus is to create an algorithm that can be used in conjunction with existing routing protocols as opposed to replacing them. This algorithm will integrate with these protocols and complement their functionality from an energy-efficiency perspective. Its design will avoid the issues that we outlined and answer the questions that we raised in Section 2.3.

2.7 Summary

In this chapter, an overview of energy efficiency issues in ad hoc networks was given. Energy models widely used in analyzing and devising ad hoc protocols were discussed. The sources of energy consumption that pertain to communications in ad hoc network were shown to exist in four main modes of operation: transmitting, receiving, idle and sleep modes. The sources of energy consumption overhead such as idle condition, collisions and protocol control messages have been discussed. The metrics used for energy-efficiency strategies have also been explored briefly. We presented a case study which sheds light on some of the energy inefficiency issues encountered in ad hoc networks. We concluded the chapter by outlining the objectives of this study based on the issues that we presented earlier.

Chapter 3: Energy Conservation Schemes

3.1 Introduction

Several techniques have been developed to address the energy efficiency issues in ad hoc networks. These techniques differ in the methodology as well as the layer of the protocol stack at which they function [25]. Energy conservation mechanisms can be classified along different lines depending on the area of focus. For example, we can classify them as geography-dependent versus geography-independent, synchronous versus asynchronous, etc. Our focus in this research is devoted to energy conservation mechanisms in conjunction with existing routing protocols. From this point of view, we can classify energy-efficient algorithms as routing and non-routing algorithms. The routing algorithms category includes both routing algorithms that were created with a strategy to address energy efficiency issues as well as energy efficient schemes that are centered around the routing strategy of existing routing protocols. Non-routing algorithms, on the other hand, are algorithms that were created to address energy efficiency issues, but they work with existing routing algorithms without directly affecting their routing strategy. Algorithms belonging to the latter category may operate at the MAC layer level, network layer level, or in between. In the following sections, we cover the most significant algorithms that belong to these categories. We discuss their characteristics as well as their limitations

3.2 Energy-Efficient Routing Algorithms

In this section, we review some of the energy-efficient routing algorithms. This includes routing algorithms, or algorithms that affect the routing strategy to achieve better energy efficiency.

3.2.1 Flow Augmentation and Flow Redirection Energy-Conserving Routing

In [8], the authors propose two algorithms with the aim of extending the network lifetime via optimizing the routing from an energy consumption perspective. Their solutions are targeted toward networks with static or slowly changing topology. They define the problem as maximizing the minimum lifetime of all nodes. The goal is to find the best link cost function which will lead to the maximization of the system lifetime. Their first algorithm, the flow augmentation (FA) algorithm, is based on creating a link cost function. This function considers the following parameters: energy cost for unit flow over the link, the initial energy and the remaining energy at the transmitting node. A good candidate for the selected path should consume less energy and should avoid nodes with low remaining energy. Since both goals cannot be achieved together, the link cost function is such that when the nodes have plenty of remaining energy, the energy cost is emphasized, while when the remaining node energy becomes small, the remaining energy parameter should be more emphasized. The path cost is the summation of link costs along the path. The algorithm can be implemented with any existing shortest path algorithms and the authors have implemented it using the distributed Bellman-Ford algorithm.

The second algorithm, the flow redirection (FR) algorithm, is based on the following idea. If we have multiple sources and destinations, then under the optimal flow (i.e.

minimum lifetime over all nodes is maximized) the minimum lifetime of every path from the source to the destination is the same. When the minimum lifetimes of the paths to the destination are not all identical then there is a set of paths whose minimum lifetime is shortest. The minimum lifetime of this set of paths can be increased by redirecting an arbitrarily small amount of flow to the paths whose lifetime is longer than these paths such that the minimum lifetime of the latter path after the redirection is still longer than the system lifetime before the redirection. The distance comparison part of Bellman-Ford algorithm is modified to obtain the shortest length paths (where the shortest path is the path that has the node with minimum lifetime of all nodes) in a distributed manner. The algorithm should also calculate the longest path which is the one with biggest capacity. This one will be assigned more flows than any other path in order to balance its minimum lifetime with those other paths. The FR algorithm was shown to have best efficiency (FR system lifetime/optimum lifetime) of 1/3, and it can be arbitrarily worse depending on network size. The authors showed through simulations that their algorithms perform better compared to minimum transmitted energy algorithms. Their simulation study, though, was done in very limited conditions: small area, small number of nodes, no consideration for idle energy consumption, and no indication of protocol performance aspect other than from a system lifetime perspective.

3.2.2 Power-Aware Routing

In [42], the authors explore power-aware metrics to use with routing protocols on top of their MAC power savings protocol, PAMAS [37]. They indicate that the strategy followed by the different routing protocols that are not power conscious would lead to fast depletion of battery power and hence quick degradation of the network operation. We have already discussed these metrics in Section 2.4. The authors implemented the first and fourth metrics (minimize energy consumed per packet and minimize cost per packet, respectively). In their simulations, the authors used sparsely populated networks and they did not consider mobility in their simulations. Their reason behind not using mobility is that the evaluation is done for power management and not routing. In our view, mobility has a considerable effect on the performance of power efficient mechanisms. For example, mobility constantly invalidates a node’s knowledge of the power management modes of its neighbors as they keep changing, which makes it more challenging to make decisions as to when to communicate with a neighbor and when to wait until it awakes. The authors do not seem to have considered idle energy consumption in their simulations either. For the fourth metric, minimizing cost per packet, the authors experimented with a linear and quadratic cost functions based on some battery discharge curves. The simulations were run with PAMAS [37] enabled which gives a source of power consumption reduction. The authors show an added improvement of 5-15% on top of what PAMAS offers. The results also show that the improvements are best when the load conditions are moderate and are negligible in case of low or high load conditions.

3.2.3 Maximum Battery Life Routing

In [46], a power-aware routing protocol that distributes power consumption evenly over nodes and minimizes the overall transmission power is proposed. This protocol uses the conditional max-min battery capacity routing (CMMBCR) scheme. It uses battery capacity instead of a cost function as a route selection metric. When all nodes on some possible routes between a source and a destination have sufficient remaining energy above a certain value, γ, the route with the minimum total transmission power (MTRP) among these routes is chosen. If all routes have nodes with low battery capacity, routes that include nodes with the lowest battery capacity should be avoided to extend the lifetime of these nodes. If the value of γ is zero, the CMMCBR reduces to MTRP. If the value of γ is equal to the maximum (100), the CMMBCR scheme reduces to the Min-Max battery cost routing (MMBCR) scheme. Simulations show that a trade-off between fairness and maximizing lifetime of most nodes in the network has to be achieved, as achieving both goals simultaneously is not possible. The author asserts that adjusting the value of γ will influence which of the two goals is reached: high values of γ achieve fairness while low values achieve extended node lifetimes. We argue though that fairness achieves longer network lifetime as it delays network partitioning, and hence from this point of view, the two goals are not mutually exclusive. It is also not clear what the recommended value of γ is from the study and it seems that the author leaves it to the user to determine based on the need. If the need is the differentiator, we argue that the user can choose one of the original schemes (MTRP or MMBCR) rather than a new one with a speculative value of the dominant parameter which may push the results in a direction opposite to what is desired.

3.2.4 Energy Drain Rate Based Routing

In [27], the authors propose route selection mechanisms for routing protocols based on a new metric, the drain rate. They propose the Minimum Drain Rate (MDR) mechanism which incorporates their new metric into the routing process. They also introduce the

Conditional Minimum Drain Rate (CMDR) as MDR by itself does not guarantee that the total transmission energy is minimized over a given route. CDMR attempts to enhance the nodes and connections lifetime while minimizing the total transmission energy consumed per packet. Each node monitors the energy consumption and calculates the drain rate (DR) for every sampling period of T seconds. The drain rate is calculated using an exponential weighted moving average method. The cost function for a node is calculated as the ratio between the remaining battery energy and the drain rate. The route is then chosen based on the highest lifetime value of the different paths where max lifetime of a path is calculated as the minimum value of the cost function over the path. Due to variations in energy drain rate over time, MDR requires the underlying routing protocol to periodically obtain new routes that take into account changing energy states of the network. Therefore, if used with an on-demand routing protocol, the protocol will have to perform periodic route discovery even if there is no route breakage. This requirement, in our view, presents a significant drawback and undermines any potential usefulness of the proposed scheme as it introduces a huge amount of unnecessary traffic in the network via flooding it with route requests and resulting replies. This affects the bandwidth, may result in severe collisions and will waste energy unnecessarily. This is particularly wasteful in the situation where the topology is static or changes slowly. The Conditional MDR (CMDR) chooses a path with minimum total transmission energy among all paths constituted by nodes with a lifetime higher than a certain value. When no route matches this condition, MDR is used. The authors used the DSR routing algorithm [24] to implement their algorithm. They used the energy model of [15] with modifications to the parameters. They did not include some of the model’s elements and they did not include the idle energy in their calculations of the drain rate. Instead of idle energy, they included energy consumed in overhearing only. We consider this to be a source of significant inaccuracy in their method that would affect the results especially in the case of sparse networks. Amongst other modifications to the DSR algorithm, they had the source node periodically refresh its cache and trigger a route recovery process every 10 seconds to better reflect the energy condition of all nodes. In the simulation results, when overhearing is considered, their algorithm shows no improvements over DSR alone, which is intuitive since energy consumed in overhearing (and idle mode) presents a source of balance by itself. In their results with no overhearing (which is not realistic), however, it seems that one of the other algorithms under comparison such as MTPR and MMBCR perform better in each of the different comparison aspects. It is not clear if they applied DSR with their modifications, especially the periodical flooding, to the other schemes that they are using in their comparison as well. If this was the case, that would skew the results to their advantage as some or all of the other schemes may not need such excessive flooding that their scheme requires.

3.2.5 Power-Aware Virtual Base Stations

In [40], the authors propose a modification to their virtual base stations (VBS) protocol of [19] to make it power aware (PA). The VBS protocol is a cluster-based protocol in which a virtual base station back bone is formed. Each mobile node selects its VBS based on the smallest node ID it becomes aware of. In the VBS scheme, any node that receives a request to be a VBS of the requesting node just accepts the request. In the PA-VBS scheme, this acceptance is conditional on some energy-constraints. The energy possessed by the requestor should be less than that of the potential VBS. The energy of the potential VBS should also be above a certain threshold. Also, if the requestor currently has a VBS, the energy of the current VBS should be below that threshold. The authors show that their PA-VBS scheme achieves energy balancing that is clearly better than that of the original VBS scheme. The PA-VBS scheme does not offer a solution to reduce idle energy consumption. It also seems that the authors have not included idle energy consumption in their experiments which affects the energy balance results.

3.2.6 Localized Power-Aware Routing

In [44], localized power aware routing algorithms are devised on the assumption that each network node has accurate information about the location of its neighbors and the destination node. This could be the case in static networks or ones in which a strong location update scheme is utilized. Nodes exchange location information via control messages. Three algorithms are proposed: power-efficient routing, cost-efficient routing and power-cost efficient routing algorithms. In the power-efficient routing algorithm, each node decides to forward packets that are intended for a certain destination to a neighbor based on the minimum transmission power between this sending node and its neighbors. In the cost-efficient routing algorithm, the node chooses the neighbor to send to, if the destination is not within reach, based on a cost function that can consist, for example, of the sum of the cost of sending to this neighbor plus the estimated cost of the route from the neighbor up to the destination. This latter part of the cost is assumed to be proportional to the number of hops in between. The power-cost efficient routing algorithm uses a combination of the two above metrics, in the form of either the product or the sum of these metrics. The authors ran their experiments on networks with high connectivity. In their evaluation, the authors showed all their methods to have limited success with large area sizes. Therefore, they modified them to get nodes to forward to neighbors only if they are closer to destination. This increased delivery rate to around 95% from highest of 59% before modification. Their results showed that nodes with the cost-efficient and power-cost efficient methods last longer than with the power-efficient method. And of all methods, the power-cost routing methods provide highest energy savings. The authors do not seem to have included idle energy consumption in their experiments, which would certainly affect the accuracy of the results especially for the cost-based algorithms.

3.2.7 Distributed Power Control Routing

In [2],[3], a distributed power control (DPC) mechanism to improve energy efficiency is proposed. It operates at the routing layer level. It estimates the amount of energy that is needed for reliable communications over any link. This transmit power information can then be used to determine lower energy paths when looking for a route. That is, transmit power is used as the link cost function in the path discovery and selection. This scheme is built on the assumption that the transmit power is the dominant source of energy consumption. Idle energy consumption was ignored in the discussions of this scheme. The scheme requires that the different nodes can record in a suitable packet format field the power level that was used to transmit the packet. It also requires that the radio transceiver estimates the power used to receive the. With the estimate of these two quantities, a node can estimate the link attenuation. Therefore the calculated transmit power for a link would be the difference between the transmitted and received power plus the minimum power level required for correct packet reception plus a security threshold that takes into account channel and interference power level fluctuations. This last quantity should be set as a function of network density, terminal speed and channel conditions. It is not clear to us from the study how would this function be structured or what weight would the different factors have in its structure. Based on the estimated power transmission associated with each link, the routing algorithms can select the packet path that results in energy saving at the end-to-end level. While implementing it with AODV, the authors have modified AODV in such a way that a destination would reply to all RREQs received within a certain time. This is to support their scheme which requires all possible routes that may connect the source and destination so that they can get the best route from energy perspective. This, in our view, can introduce additional significant traffic that can offset any benefit their algorithm may be able to introduce. In addition, the scheme ignores the power level that the different nodes have and only focuses on transmit power needed per link which can result in energy imbalance. To evaluate the scheme, the authors ran the simulations for relatively light density, small area and light load conditions. Their simulations showed improved energy performance over the routing algorithms without their modification, without much degradation in protocol performance, in the static case. With mobility and especially with AODV, their simulations showed that the performance of the routing protocol degrades significantly when applying their scheme.

We notice from the previous discussion that most of the routing-related energy-efficient algorithms focus mainly on the balance between the different routes and do not take into consideration idle energy consumption. The existence of the idle energy introduces an undesired source of energy balance between nodes. Therefore, including idle energy is crucial in the discussion as it is likely to show that the effectiveness of such algorithms is limited in the absence of a mechanism to reduce or eliminate idle energy consumption.

3.3 Energy-Efficient Non-Routing Algorithms

In this section, we discuss some of the energy-efficient non-routing algorithms. These are the algorithms that focus on conserving energy but without interfering with the functionality of the underlying routing algorithm.

3.3.1 IEEE 802.11 MAC Power Management

IEEE 802.11 MAC [50] includes power management capabilities that attempt to reduce power consumption of network nodes. The idea behind IEEE 802.11 power management [41] is to switch off the node’s transceiver whenever it is not needed. Since there is no way to predict in advance when the transceiver should expect to receive packets, it has to wake up periodically. Two power management states exist: sleep and awake. Data get buffered in the senders when the receivers are in the sleep state. In ad hoc mode, all nodes in the network are synchronized to wake up at the beginning of each beacon period. Messages are sent during the period when all nodes are awake. Data transmission is done by signaling the existence of data that needs sending inside a small interval called ATIM at the beginning of each beacon interval. During the ATIM interval, if a node has data to be sent to other nodes, it sends an ATIM message indicating the receivers of this data. If any node receives a message indicating that there is data for it, it acknowledges and stays awake for the rest of the beacon interval. Other nodes that do not receive such a message can go to sleep after the ATIM window is over. Nodes that have buffered data can then transmit this data during the rest of the beacon interval. This power management scheme has some drawbacks. For example, it requires a complete synchronization of all network nodes to wake up at exactly the same time. It can lead to severe collisions in case of heavy traffic networks especially during the ATIM window. This could lead to severe delays and reduced throughput. This calls into question the scalability of this power management scheme.

3.3.2 PAMAS

PAMAS [37] is a multi-access protocol for ad hoc networks. It is based on the MACA protocol [26] but its functionality relies on the existence of a separate signaling channel on which RTS/CTS packets are transmitted. For power conservation, PAMAS requires nodes to shut themselves off if they overhear transmissions over the data channel. Also, if a node does not have packets to transmit, it should power itself off. If there was packet transmission going on while the node is going to sleep, it powers itself off for the period of this packet transmission. If it wakes up and hears packet transmissions, it sends a probing message (if its send queue is empty) asking other nodes for the length of the current transmission. Transmitters would respond with a message specifying the time at which their current transmission ends. If a node powers on with a non-empty send queue, it sends an RTS message instead of a probing one. In the cases where the nodes turn themselves off only for the exact period while neighbors are transmitting data packets, there would be no effect on packet delays. The authors have shown that PAMAS reduces power consumption by almost 50% in the case of high loads (0.5 - 3 packets/sec/node) for complete networks (fully connected networks) and below 10% for line networks. The simulations were run for networks with 10 and 20 nodes with packets sizes of 512 bytes. The authors also measured the throughput (packets/sec) transmitted in the network as a function of the load (packet/sec/node). The authors have shown PAMAS to achieve 50% of the theoretical maximum throughput for sparse networks (10 nodes) and 70% of the theoretical maximum throughput for dense networks (20 nodes). There was no indication of the corresponding results for the case without PAMAS. PAMAS operates with the assumption that when the node is idle (idle medium), no power is consumed. This limits the power conservation advantages of the PAMAS protocol.

3.3.3 Geography-informed Energy Conservation (GAF)

In GAF [47], the network is divided into virtual grids whose sizes are dependent on nodes’ nominal radio range. Each node uses its location information to determine its grid. All nodes within a particular grid have equal packet forwarding ability. Nodes can be in one of three states, active, discovery or sleeping. Nodes in a specific grid decide which node will remain awake while the rest go to sleep via a certain node ranking system that takes into consideration a node’s state, then a node’s energy level and finally node ID as a tie breaker. GAF can achieve node energy balancing within a specific grid only. This is done during the discovery period of a node that was in active state, where other nodes can take over routing duties during this period. As far as routing is concerned, GAF does not interact with the routing protocol at all, it leaves it to the routing protocol to recover from breakages that result from an active node that was involved in routing deciding to go to sleep based on GAF’s criteria. The authors also exclude traffic nodes from the forwarding functions or from even running GAF. This is unrealistic since in reality, all network nodes are expected to share routing functions and it may not be possible to know in advance which nodes are traffic nodes. In addition, if it turns out that all nodes in the network will be sources or destination of traffic at different points of the time, this would render GAF unusable. The simulation results that the authors presented were based on very aggressive conditions. Yet, their results (when we consider well known cases e.g. AODV alone without GAF) seem to strongly contradict with previous studies as well as our own findings. For example, in some of their key simulations they used maximum node speeds of 20m/s and send rates of 20 packets/sec, with data packet sizes of 512 bytes. The simulation area was 1500m × 300m, which is the same as the simulation area size used in our simulations and other studies that we are referring to ([6] and [35]). Under these conditions one would expect congestion to affect normal protocol performance severely as we have experienced in our own simulations and as has also been indicated in these previous studies. This was not the case in their study. The authors maintained throughout the study that traffic load levels do not affect the results, a claim that is rather odd in our view. However, according to their simulations and under the above mentioned conditions and different pause times, GAF seems to extend network lifetime to a degree that depends on the mobility level (pause time). The authors reported that after the end of their simulations, 30-40 % of the nodes were still alive while with AODV alone, all nodes ran out of energy mid-way through the simulation. They found that the shorter the pause time, the longer the network life time. They attributed this to the fact that higher mobility results in better load balance. The energy savings have been shown to be 50-60 % over AODV alone. As for PDR, with a lower send rate (10 packets/sec), GAF has noticeably worse PDR than AODV alone and this can reach as low as 85% versus 95% for AODV alone in high mobility conditions. The average latency can reach 0.35 seconds versus 0.16 seconds for AODV alone.

3.3.4 Energy-Efficient Coordination for Topology Maintenance (SPAN)

In SPAN [10], routing duties are performed by "coordinator" nodes. A node's decision to become a coordinator is based on the amount of remaining energy it possesses and the number of pairs of neighbors it can connect together. Therefore, coordinators stay awake continuously and perform multi-hop packet routing within the network while other nodes remain in energy saving mode and periodically check to see if they should wake up and become coordinators. SPAN requires modifications to the route lookup process at each node. At any time only those entries in a node's routing table that correspond to currently active coordinators can be used as valid next hops. This implies potentially pervasive changes for some routing protocols. As far as energy fairness among network nodes is concerned, we have comments on SPAN similar to what we mentioned about GAF. The energy balance that SPAN can perform is based on local conditions in the coordinator neighborhood. A candidate node for becoming a coordinator now that gets actually selected for this role may be forced to remain a coordinator for an extended period of time due to network topology changes. Also, since coordinators need to stay awake all the time, this will result in wasting their energy when there is no traffic to be handled. In their simulations, the authors organized the nodes in such a way that the traffic nodes never move and are not involved in forwarding. Therefore it seems like there were excluded from the normal function and behavior of the rest of the nodes, as in the GAF case. The authors have experimented with traffic rates of 3 packets/sec with data packet sizes of 128 bytes. To examine the effect of packet send rate on SPAN's performance, experiments showed that with no motion, packet delivery rate started to suffer noticeably at packet send rates of 4 kbps. To measure the effect on packet delivery latency, the authors ran simulations with no motion and send rates of 3 kbps and changed the simulation area size. The results varied with the size of the simulation area. As an example, with an area of 1000m ×1000m, the delay was 40.5 ms with SPAN versus 16.9 ms for the underlying geographic forwarding routing protocol alone. As far as energy savings are concerned, at the end of the simulation, with SPAN enabled, the percentage of the remaining energy was 60% versus 20% without SPAN. It is not clear what level of mobility, if any, was used for these energy saving calculations. With a 500m × 500m area of simulation, they showed that SPAN extends network life time by almost a factor of 2. Again, it is not clear what level of mobility was used to obtain these results.

3.3.5 Asynchronous Wakeup Scheme for Ad Hoc Networks

In [48], the authors devised an asynchronous wakeup schedule for mobile ad hoc networks. In their design, the authors rely on the existence of sufficiently overlapping active time slots between communicating nodes. Their algorithm is composed of two main procedures: neighbor discovery and neighbor bookkeeping. The neighbor discovery procedure is based on each node selecting its own active and sleeping time slot with a wakeup scheduling function common to all nodes. At the beginning of the active slot, a node transmits a beacon message that has its ID and other scheduling related information. The authors indicated that the algorithm used will ensure that two neighbors will be able to have overlapping active periods which will ensure that they hear each other's beacon messages. Each node keeps a neighbor list that has neighbor information such as ID, clock, schedule, etc. When a node hears a neighbor's beacon message, it knows that it will remain active for a specific known period of time and can therefore transmit any data that it has buffered for it during this active period. This may not be sufficient due to collisions that beacon messages may suffer. Therefore, and in low mobility scenarios, the bookkeeping procedure is beneficial in helping a node in keeping track of neighbor's schedules in order to determine its neighbors' wakeup schedules. The authors examined their protocol by simulating it using a greedy geographical routing protocol that they implemented. They used two power management algorithms with their scheme: one is slot-based which is mainly based on MAC 802.11 power management and triggers the receiver to remain awake during the next slot if there is traffic above a certain threshold queued for it. The other power management algorithm that they used is the on-demand power management algorithm [49] that we will discuss in some detail later. In the absence of power management, node discovery is done using HELLO messages with an interval of 1 second. Using a static network and on-off traffic sources, their algorithm with slot-based power management showed lower PDR than when there is no power management. The difference was 2-6% and decreased with the increase of the traffic load. Energy consumption in this case was significantly lower than without power management, (less than 0.3W versus above 0.8W). Using on-demand power management under these simulation conditions produced a PDR that is close to the PDR in the case of no power management (difference is slightly above 1%) but the energy consumption was above 0.4W. The authors also experimented with the case of mobile networks. They used node speeds of 20m/s and 10 long-lived CBR connections. With slot-based power management, the PDR deteriorated significantly by 12-15% compared to that of the case of no power management (loss decreases with the decrease of mobility). Power consumption remained at the same levels as in the static case. With on-demand power management, the PDR was actually better than without power management. This is attributed to two reasons. First, the HELLO neighbor discovery messages interval without power management is larger than with power management (1 second versus 0.7 seconds). Second, the greedy geographical forwarding protocol that was used for routing does not attempt to salvage packets for neighbors that moved away. As for energy savings with on-demand power management in the case of mobility, it was significantly less than in case of static networks (the consumption was between 0.6-0.7 W versus slightly above 0.8 W without power management).

3.3.6 On-demand Power Management

In [49], the authors propose an on-demand power management framework that integrates routing information from on-demand ad hoc routing protocols and power management capabilities from the MAC layer. Energy conservation is achieved by turning on and off the radios of specific nodes in the network, which is driven by active communications in the network. Connectivity is only maintained between pairs of senders and receivers and along the route of data communication. The transitions between power management modes are done via establishing and maintaining a soft-state timer by control and data packets in the network i.e. by communication events. The timer value is determined by the packet type. The soft-state timer is refreshed by events similar to those that trigger the move from power save mode to active mode. A node keeps track of its neighbors' power management modes either by HELLO messages or by snooping transmissions over the air. On-demand power management uses the capabilities of MAC protocols e.g. IEEE 802.11 MAC to switch power management states of nodes and buffer data for sleeping nodes. It uses routing info to decide when to turn nodes on and off. A node can be in one of two power management modes: active mode (AM) where the node is awake, or in power save mode (PS) where the node is sleeping and wakes up periodically to check for pending messages. If a certain path will be used, nodes along this path are kept awake. The commitment to a path is determined by the types of messages. Knowledge of the semantics of the messages helps with the power management decisions. For example, control messages (e.g. route request, link-state, location updates, etc) should not trigger the node to stay in active mode. Data packets on the other hand should trigger active mode. The keep-alive timer value should be set according to the packet inter-arrival times to prevent the node from going to sleep while in data transmission. This also applies to some control messages such as route reply messages which indicate that there are some packets to follow in this route. We see two issues with this strategy. First, if any delay is encountered due to collisions or other network non-fatal issues, the timer may expire prematurely leading to packet losses. The second issue is the potential significant changes to the power management algorithm with the change of the underlying routing protocol due to the need to incorporate new message semantics. The authors indicate that the way the algorithm determines the power management mode of the neighbor is through passive inference. This is done to save energy that would otherwise be consumed in HELLO messages. This works via two indicators. The first is lack of communications in a specific period of time. The second is packet delivery failure e.g. RTS retry timeout in IEEE 802.11. In our view this method of passive inference is not robust as there may be many reasons why the neighbor is not communicating other than being in power save which can lead to many issues such as unnecessary delays. The list of neighbors with their power management modes is maintained via snooping transmissions in the air when they are awake. It seems that the algorithm focuses on whether the neighbor is asleep or awake but does not seem to have a way to know when that neighbor will become awake. This can imply more delay and energy waste since the neighbor may be awake and idle while the sender is not aware of this. It is not clear to us if synchronization is required with this protocol, as it uses the IEEE 802.11 MAC power management which functions based on node synchronization. The authors ran some simulation experiments to demonstrate the performance of their algorithm. With on-off traffic load (on for 10 sec and off for 100 sec) and no mobility, the PDR started to drop noticeably at around 1500 bps send rate, which is a rather low send rate in our view. Their energy goodput (total bits transmitted/total energy consumed) increases almost linearly with traffic load according to their diagrams, at a slope higher than without their algorithm (example of the difference is 1000 bits/J at traffic rate of 1500 bps). They also examined packet delivery latency in the static case with on-off traffic of 1kbps. As an example, with one 3-hop connection, spikes of about 0.6 sec delays resulted with their algorithm versus almost zero delays without their algorithm. With mobility, simulations showed that energy savings are considerably less than in the static case. For example, energy goodput ranged between 200 – 1500 bits/J depending on the mobility (increases as the mobility decreases). PDR is also noticeably lower than the case with no power management even with relatively low send rate (1kbps). As for packet delivery latency, with 10 CBR connections and 1kbps traffic, packet delivery latency can reach 4 seconds versus almost zero without their algorithm.

We can see that each of the discussed energy-efficient non-routing algorithms has some advantages over the others and also some drawbacks. In our assessment, the SPAN [10] algorithm as well as the GAF [47] algorithm to some extent, use a deterministic way to hand over routing duties among the forwarding nodes. This determinism gets affected significantly by mobility conditions as nodes that were elected for forwarding may move away from their posts, potentially resulting in lost connections for a period of time until other nodes wake up and take over. The two algorithms also suffer from some unrealistic assumptions that were incorporated into their designs. An example of this is excluding traffic nodes from running the algorithms altogether. The On-demand Power Management [49] algorithm, on the other hand, does not make such unrealistic assumptions. It has also been adapted for testing with some of the well-known routing algorithms. Its major drawback, in our view, is the heuristic method of deciding on the sleep state of the neighbors and this seems to affect its results, especially in the case of mobility.

3.4 Summary

In this chapter we discussed some of the energy-efficient algorithms for mobile ad hoc networks. Based on the relevancy to our research, we divided the algorithms into two categories: routing and non-routing energy-efficient algorithms. The routing algorithms are the ones that are created as energy efficient routing algorithms or the ones that were created with the sole purpose of influencing the routing strategy from an energy efficiency standpoint. The non-routing algorithms are the ones which are not related directly to routing but aim at integrating with existing routing algorithms to achieve an overall energy-efficient operation. Most studies that focused on balanced energy-efficient routing either ignored the idle energy factor, or found their techniques ineffective on their own. This is because idle energy presents a source of balance. Studies that belong to the non-routing category achieved their goals with varying degrees of success, especially when considering the limitations of the algorithm at hand. Almost none of the studies of this category focused on energy fairness as a primary goal. Therefore, we see an opportunity to create a scheme that focuses on the energy fairness aspect as a primary goal. It should also cover the shortcomings that we have outlined for some of the algorithms that we discussed. This includes distributed functionality, realistic network assumptions and a robust method of determining the sleep state of the neighboring nodes.

Chapter 4: PIES - Protocol Independent Energy Saving Algorithm

4.1 Introduction

It is evident that energy efficient schemes are of crucial importance in the context of ad hoc networks. Since the goal is to save energy, and since one of the main sources of unwanted energy consumption is idle energy, the need to eliminate or reduce this energy becomes one of the main targets of such schemes. This requires the energy efficient mechanism to introduce some arrangement that includes sleep periods of the network nodes' wireless interfaces to lower this unwanted source of energy consumption.

The introduction of such schemes has its challenges. For example, how would a node know whether a neighbor is awake so that it can communicate with it? How can it make such determination without introducing significant amount of traffic to the already bandwidth-limited networks? To answer these questions, we summarize the main requirements that should be fulfilled by an ad hoc energy conserving mechanism in the following points:

- The algorithm should not have a significant impact on the network operation such as throughput.
- It should not introduce sources of significant energy consumption to the network.

- It should not introduce sources of significant traffic to the network.
- It should have a robust method of determining the mode of operation of the neighbors.
- It should be fair. Network nodes should be treated equally. This both provides the same level of service to network clients and prolongs network lifetime.
- It should be distributed. This avoids having single points of failure, bottlenecks and energy unfairness.

In this chapter, we propose a new energy conserving algorithm [17]. This algorithm is intended to work in conjunction with existing ad hoc routing protocols of the different categories. It addresses the above requirements and others as we will indicate. We call this algorithm *PIES,* which stands for "Protocol Independent Energy Saving". This name reflects the nature of the algorithm. Its design is modular and can be used by the different routing algorithms without making changes to it. This is because, by design, the goal is to make the algorithm independent of the nature and semantics of the routing protocol. The routing protocol would interact with PIES to incorporate the energy efficiency services that it provides. We have also ensured in the design, as much as possible, that PIES in turn does not impose core design changes to the routing protocol. The description of the design and elements of PIES is covered in the following sections.

4.2 PIES Design Philosophy and Goals

As we have shown earlier, idle energy consumption constitutes a significant percentage of the overall energy consumed by the wireless interfaces of network nodes. Therefore, reducing this energy should be a cornerstone in any energy conservation efforts. As will be seen, our proposed algorithm, PIES, addresses the issue of idle energy consumption in a manner fair to all network nodes. Different nodes are given equal opportunities to conserve idle energy. When idle energy is addressed, another factor remains that may still affect energy fairness within the network. This is explained as follows.

Since ad hoc network nodes also assume the role of traffic routers, some nodes may need to cooperate in order to direct traffic that may not have been intended for them in the first place. Many routing strategies aim at finding the fastest and shortest routes for the traffic between two nodes that need to communicate. This may penalize some nodes that happen to be in a location that causes it to be part of several optimal routing paths. As an example, consider the network topology of Figure 4-1.

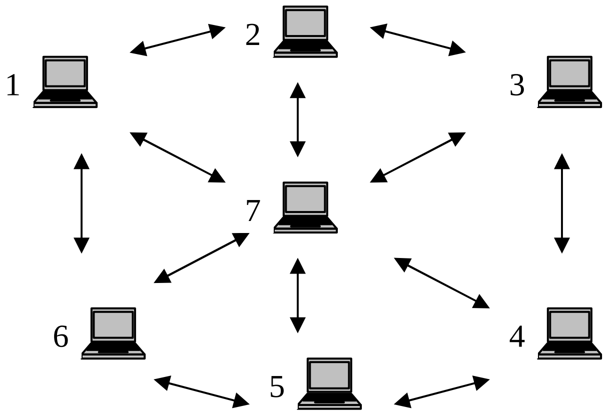


Figure 4-1 : Topology with a node in the middle on all shortest routes for the network

If the decision is to use the shortest path, node 7 would be the obvious choice when routing data between node pairs (1,4), (2,5) and (3,6). This will always be the case as long as node 7 is alive, despite the existence of other routes. For example, if we want to route packets between nodes 2 and 5, and assuming that all adjacent nodes are within radio range of each other, we can use the routes (2-1-6-5) or (2-3-4-5), in addition to (2-7-5). The problem with continuously using node 7 for packet routing is that it will run out of energy much faster than other nodes in the network. From an energy point of view, this causes two issues:

- Nodes are not treated as equal, which means that some nodes will run out of energy faster than the others due to their strategic location, thus causing them to cease to serve their own users faster than others. This presents a ‘local’ problem that affects node users.
- Network partitioning may occur. Since some nodes may be critical for routing between certain nodes at some point of time, if these critical nodes run out of energy, routing between these nodes can no longer be done. This presents a ‘global’ problem that affects parts of the network or the whole network depending on the case.

The PIES algorithm helps address these issues. It provides the underlying routing protocol with the capability to make and implement routing decisions that take into consideration the energy state of the nodes that can be used for routing traffic. This can transform the existing routing algorithm into an energy-conscious one. This strategy helps to maximize the lifetime of network nodes and hence the network operation as a whole [7].

The main goals of the PIES algorithm can be summarized as follows:

- Fair energy conservation via:

  - Rotating sleep periods equally among network nodes thus giving nodes equal opportunity for reducing energy consumption
  - Assisting routing algorithms in making routing decisions based on energy fairness
- Little impact on network operation, for example, PIES introduces slight or no additional traffic or energy cost
- Distributed processing of the algorithm which ensures robust operation that is not affected by the failure of one or more nodes
- Modular nature which facilitates integrating it with existing routing algorithms

4.3 PIES Algorithm Functional Description

From a functional point of view, PIES can be considered to consist of two main units. One of these units handles the energy conservation operation. This is done through managing the nodes' sleep and wakeup periods. The other unit or aspect of the algorithm takes care of supporting the routing protocol, as far as energy management decisions are concerned. It helps to ensure the routing protocol makes routing decisions that serve a specific goal. For example, whenever possible, nodes carry out routing duties that are proportional to their energy levels compared to each other. Figure 4-2 shows the interactions of PIES with the network and MAC/PHY layers.

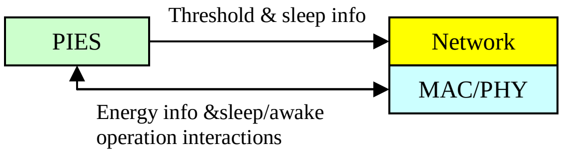


Figure 4-2: PIES interactions with the network and MAC/PHY layers

4.3.1 PIES Energy Consumption Management

When PIES is enabled, nodes operate in one of two main modes: “*asleep*” or “*awake*”. In order to ensure maximum energy fairness, PIES enforces a configurable two-step cycle of operation for each node of the network. One portion of the cycle is a *mandatory sleep* period while the other portion is a *mandatory wakeup* period. While awake, a node can communicate normally with other nodes as per the routing protocol that is in use. The length of the mandatory wakeup period affects the connectivity of the network as nodes establish their knowledge of the current neighbors and network conditions during this period. While “asleep”, a node cannot exchange data with the external world. The only exception is sending broadcast messages that pertain to the routing protocol. In this case, the node goes to what is called “*sleep-pending*” state, where it wakes up for the duration of the broadcast and then it goes back to the “asleep” state to resume sleeping for whatever is left of the current sleep period. The reason behind not allowing unicast traffic and allowing only the sending of broadcast traffic is as follows. If we are to allow the node to send unicast messages during the sleep period, the node will have to abort the sleep and make sure it remains awake for the period of exchanging the control messages (RTS, CTS and ACK) as well as the data with the other end. This will not only complicate the operation of the algorithm, but will also potentially deprive the node from having a decent sleep period since the amount of time taken by this exchange can be unpredictably long in cases such as transmission errors. This contradicts with the goal of fair energy conservation for all nodes. However, we still allow broadcast messages to be sent by the sleeping node since the interruption of the sleep mode will be minimal in this case and at the same time this permission will help preserve healthy operation of the routing algorithm. Figure 4-3 gives a description of how transitions between modes occur.

While the node is asleep, other nodes may have some traffic that they need to send or forward to it. In order to make sure that this traffic will not be lost, nodes need to know when the node in question will start its sleep period and when it will become awake again which in turn determines if the node is asleep or awake. With this knowledge, the sending nodes buffer the traffic going to the sleeping node during its sleep period and then release it when it wakes up. This can be achieved via going-to-sleep and waking-up notifications that the node can send to the outside world upon going to sleep and waking up, respectively.

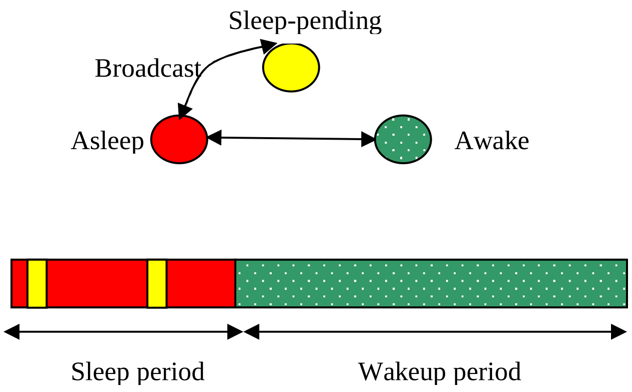


Figure 4-3: PIES modes of operation

However, this method, besides being unreliable, introduces an additional energy and traffic cost to the network operation. Alternatively, PIES relies on each node being aware of the sleep status of any node in the network via the following formula:

$$S_c = \{[\mathit{int}((\mathit{NOW} - S_p)/(W_t + S_t)) + 1] \times (W_t + S_t)\} + S_p, \qquad (4\text{-}1)$$

Where:

S_c is the end of the current, if currently asleep, or next sleep period

S_p is the end of any known previous sleep period of the node

W_t is the PIES mandatory wakeup period in seconds

S_t is the PIES mandatory sleep period in seconds

NOW is the current time

This formula can be derived as follows:

Let us assume that a node ends its sleep time at some point of time, S_m. It wakes up for W_t and it then sleeps for S_t and the cycle is repeated. Therefore the total cycle length is (S_t + W_t), or for simplicity, T.

Therefore, to get the next sleep end time, S_{m+1}, we use the following equation:

$S_{m+1} = S_m + (S_{m+1} - S_m)$ and,

$$S_{m+2} = S_{m+1} + (S_{m+2} - S_{m+1})$$
$$= S_m + (S_{m+2} - S_m)$$

and so on.

To generalize, if we want to calculate the next sleep end time S_n using any sleep end time S_m where $S_n > S_m$, the equation to use is:

$$S_n = S_m + (S_n - S_m)$$

$= S_m + [(S_n – S_m)/T] \times T$

$= S_m + [(S_{n-1} – S_m)/T + T/T] \times T \qquad (4\text{-}2)$

where $S_n = S_{n-1} + T$

At time, t, where $S_{n-1} \leq t < S_n$, to be able to get S_n using (t) and any known sleep end time S_m, we proceed as follows.

Since $S_{n-1} \leq t < S_n$ and $t – S_{n-1} < T$, therefore $\{(t – S_m)/T – (S_{n-1} – S_m)/T\}$ is positive real number that is less than 1.

Therefore, the integer portion of $(t – S_m)/T$ is equal to $(S_{n-1} – S_m)/T$ or,

$\mathrm{int}[(t – S_m)/T] = (S_{n-1} – S_m)/T \qquad (4\text{-}3)$

Therefore, and since we know t but not S_{n-1}, (4-3) can be substituted in (4-2) to get the following:

$S_n = S_m + [\mathrm{int}((t – S_m)/T) + 1] \times T$

$= [\mathrm{int}((t – S_m)/(S_t + W_t)) + 1] \times (S_t + W_t)] + S_m,$

which is equation (4-1).

By determining the end of the current sleep period (if the node is asleep) or the next one (if it is awake), the sending node can determine whether it can send traffic to the node or if it should buffer it and send it later when it becomes awake. This formula makes it possible for any node to know the sleep state for any other node that it is aware of, regardless of the mobility conditions of the network. If no prior sleep period of the node is known, the initial sleep period can be used. The first or initial sleep period for all nodes can always be a function of a known unique parameter of the node. This mechanism assumes that there is no time shift between the two communicating nodes and we have based our experimental simulations on this assumption. This can be the case in coordinated operations that rely on the participants having the same time knowledge. It can also be the case where the different nodes have their clocks synchronized with one common external time source [45]. However, if a time shift exists between two communicating nodes, the time of each of them can be piggybacked on routing protocol messages. As data will only be sent after a route has been discovered, this clock shift can then be factored into equation (4-1) and a slight slack time can also be introduced in the beginning and end of the sleep time as an extra precaution. It is worth mentioning that PIES maintains a table at each node that we call *the PIES table* which includes information about network nodes sleep period and energy info for easy lookup. This table gets updated periodically and upon any new relevant knowledge that the node obtains to make sure the information stays fresh. Once a node needs to communicate with another node, it uses its own PIES table to obtain the sleep period and energy info of that node. If it does not find the sleep info in the table, it can use the above formula to determine the sleep status of that node. This method allows complete predictability of the other nodes' sleep status in a way that is robust and at the same time does not introduce traffic or energy consumption costs to the network operation.

Despite the fact that several nodes of the network may be asleep at the same time at certain points of time, we have found it important to make sure that not all nodes of the network have fully overlapping sleep periods (that start and end at the same time). For that reason, PIES uses a *sleep separation* parameter which represents a factor in determining the start of the initial sleep period of the node. The sleep separation factor can be of a constant value or a function of some parameter. In our experiments, we used it as a function of the sleep time. It is important to choose the sleep separation factor as well as the sleep time/wakeup time combination in such a way that it prevents two nodes from having totally out-of-sync schedules. As a measure to ensure that, we recommend avoiding the use of sleep time and wakeup time of equal length. However, we experimented with this case in our simulations to test the impact. For simplicity, we selected the sleep separation factor in our simulations to be 10% of the sleep time. We then calculated the initial sleep start delay of a node to be the sleep separation factor times the node ID (with nodes being numbered by integers, starting at 0).

So, in summary, based on the node ID (which has to be known for routing purposes), each node in the network can deterministically determine whether a node is awake or asleep.

With energy consumption management as described above, it is possible to limit idle energy consumption to lower levels as we have effectively condensed the sending and receiving activities in shorter time periods, reducing energy consumption by putting the nodes to sleep for the rest of the time. The only visible side effect of this strategy is the possible impact on latency, which also depends on the network traffic.

4.3.2 PIES Energy Balance Management

The PIES algorithm also interacts with the routing algorithm to balance energy consumption that occurs during the routing operations. It does so by helping the routing protocols to balance the routing duties in a fair manner such that no node will be over-utilized while others that can lead to the same destination remain under-utilized. For this to happen, there needs to be some means for signaling the need for a switch from one route to others. For this purpose we use what we call the PIES "*rotation threshold*".

4.3.2.1 PIES Rotation Threshold

Many routing algorithms' selection of the routing path is based on the number of nodes on the path. This helps keep latency at a minimum under normal conditions. Therefore and to try to both balance energy consumption and at the same time maintain an acceptable degree of packet latency, the criteria to be used for switching nodes has to be considered carefully. For example, node switching should not occur unless there is a considerable difference between node energies. For this purpose, PIES uses energy thresholds, or rotation thresholds, to help the routing algorithms make this decision. The initial value of the rotation threshold can be preset to a certain percentage of the node's initial energy, say 80%, as an example. However, once the initial threshold is reached by any of the nodes, a new value of the threshold needs to be set for this node and announced to its neighbors. There are several possibilities for setting new threshold values. One way is to use *constant* decrements. For example, if the initial value of node energies is 100 Joules, the threshold will be decremented by 20 Joules every time it needs to be updated. Another way of doing this is by setting the threshold *linearly*. That is, we decrease with a certain factor of the previous value of the rotation threshold. For example, if the current value of the threshold is "e_1", the new threshold, e_2, would be:

$$e_2 = \rho \times e_1, \tag{4-4}$$

Where "ρ" is the *rotation* factor. The linear approach allows us a finer granularity in rotating the routing duties. However, once we reach a certain small threshold value, it would become more efficient to decrement thresholds in constant values to avoid threshold messages getting sent too often, if PIES was configured to send such messages. Announcing new threshold values can be done via either sending PIES specific messages to neighbors or piggybacking this information on other packets the node sends using a PIES specific header that gets populated upon sending the packet. The method is configurable and it depends on the underlying routing protocol. It should be noted, however, that even if the chosen method is PIES' specific messages, the cost is still minor as the messages are sent only when a threshold changes which generally does not happen frequently.

4.3.2.2 Routing Protocol Energy Management Interactions with PIES

There are different points of interaction that can exist between PIES and the routing protocol. The routing protocol needs to interact with PIES to determine node sleep state and hence forward delays. In addition, the routing protocol can make energy-conscious decisions using the data provided by PIES. It is worth mentioning that a variety of decisions can be made based on this data. For example, the routing protocol can use the route that has nodes with highest average overall energy, based on the different energy rotation threshold levels.

Since our goal is to ensure maximum energy fairness, we have focused on experimenting with having the routing algorithms differentiate between routes based on the minimum rotation threshold values of the nodes on the route. Therefore, when the routing protocol has the choice between several routing paths, the decision is made based on minimum threshold values on these paths. This requires the interaction between the routing protocol and PIES upon making this decision to obtain the necessary data. This implies slight modifications to the routing protocol, to involve some PIES APIs if PIES is enabled. These APIs then trigger the necessary calculations and enable the routing protocol to decide accordingly.

4.4 Summary

In this chapter, we presented our proposal for energy conservation in ad hoc networks. We first outlined the requirements that should be fulfilled by an energy conserving algorithm. We then presented PIES, which is an energy saving algorithm that works in conjunction with existing ad hoc routing algorithms. We described the design strategy of PIES and we presented its two main components. PIES has a distributed mode of operation that aims at achieving fair energy conservation for all network nodes. It does so in two steps. First, it ensures that all nodes equally share in the energy conservation mode and hence achieve an equal amount of idle energy reduction. Then it helps the routing protocol with energy-aware routing decisions via making the energy conditions of nodes along the different paths available. PIES ensures packet delivery by allowing network nodes to determine the sleep state of neighbors with certainty. This way, they manage their communication with neighbors based on their proper mode of operation. By design, PIES does not introduce a significant cost, both from a traffic and an energy point of view, to the network operation.

Chapter 5: PIES Performance Evaluation

5.1 Introduction

In the previous chapter, we introduced PIES, which integrates with existing ad hoc routing algorithms and helps with overall balanced energy conservation. In this chapter, we introduce the results of evaluating the PIES algorithm performance from different perspectives. We first assess its performance when it is integrated with routing protocols of different categories. We then show its ability to scale with higher network population and traffic. We also evaluate its performance with different types of traffic and demonstrate the effect of changing its parameters on performance. We then show the effect of using PIES on enhancing network lifetime. Finally, we perform a comparison with the on-demand power saving algorithm.

5.2 Evaluation Method and Environment

In order to evaluate PIES in different conditions, we will be using simulation experiments. Simulation is one of the most important methods in evaluating networking related algorithms and protocols. The use of simulation has many benefits that make it a very powerful tool [1],[22]. For example, it allows repeating scenarios, experimenting with different parameters individually and investigating a variety of parameter combinations. It also allows us to understand and rectify the behavior of an algorithm or a system in conditions in which we may not be able to easily experiment with the scenarios of interest in the real environment.

To achieve steady-state performance results, we have to run simulations for long enough. Compared to experimental set-ups reported in the literature, and based on the fact that we use relatively high node motilities, we determined 600 seconds as being sufficient for this purpose.

In these evaluation experiments, we use the ns2 simulator [13] to conduct our performance evaluation. This includes the CMU Monarch Project's [12] wireless and mobility enhancements to ns2. In the following, we describe the simulation conditions that we generally used in our simulation experiments. Unless otherwise specified while discussing a certain experiment, these are the conditions under which the experiment was conducted. We ran our simulations for different pause times at a selected high maximum speed to test the impact on PIES operation. This represents different mobility conditions of network nodes. Each data point is an average of five simulation runs with different randomly generated mobility scenarios. For all runs, we used identical traffic models. We used the same mobility and traffic scenarios across all the cases under comparison. Table 5-1 shows the values that we used for the simulation runs. In general, a statistical analysis based on the corresponding confidence intervals confirms the trends observed by looking at the experimental averages. We will be discussing these intervals when experimenting with the different routing categories, as examples. For the rotation threshold, we applied equation (4-4), with $\rho = 0.9$. However, when the energy reaches a certain low threshold (we selected 30 Joules), we switch to constant decrements of 15 Joules. In our simulations, threshold information is exchanged via the routing protocol control messages (piggybacking) as well as PIES announcements upon threshold changes. We have verified that piggybacking is sufficient for this purpose, however, we included the PIES overhead also in our simulations to get worst-case condition results.

Table 5-1: Simulation parameters

Number of nodes	50
Dimensions of simulation area (m×m)	1500×300
Initial node energy (Joules)	1000
Simulation time (seconds)	600
Traffic type	CBR
Number of traffic connections	20
Maximum node speed (m/s)	20

With these conditions, we ran our simulations using different values of PIES mandatory wakeup time (WT) and mandatory sleep time (ST) pairs to examine the effect of changing these dominant PIES parameters on the network operation and results. For the experiments where we wanted to check the effect of changing PIES ST/WT values, we used ST/WT values in seconds of: 0.5/0.5, 0.75/1.0 and 0.25/0.5.

As far as the evaluation criteria are concerned, we measure two main aspects: the energy performance and the network operation performance. For energy performance, we measure the energy savings and the standard deviation of node energies. The standard deviation is used as means of measuring energy fairness: the lower the standard deviation, the fairer the energy consumption across nodes is. For network operation performance, we use the packet delivery ratio and packet delivery latency as evaluation measures.

5.3 Energy Model

In our simulations, we consider energy consumption in the sending, receiving, idle and sleep modes of operation. Experiments e.g. [14],[43] have been conducted to measure and determine the power consumption patterns in the different operation modes. In our simulations we use the power consumption values for send, receive, idle, and sleep modes that were obtained through measurements in previous studies e.g. [14]. Table 5-2 shows the values that we used.

Table 5-2: Power consumption parameters

Rx Power Consumption	1.0 W
Tx Power Consumption	1.4 W
Idle Power Consumption	0.83 W
Sleep Power Consumption	0.13 W

5.4 PIES Performance with Reactive and Proactive Routing Protocols

In this section, we examine the performance of PIES with the different categories of routing protocols, namely the on-demand (or reactive) routing and the proactive routing algorithms. The intent is to ensure its ability to function properly with the different types of routing strategies and to provide an assessment of the required modifications of the routing algorithm in each case. For this purpose, we integrated PIES with the AODV (reactive routing) and the OLSR (proactive routing) routing protocols. We also show later in this chapter the performance of PIES with the DSR routing protocol, which belongs to the reactive routing category, as part of one of our comparison experiments.

5.4.1 Evaluation Results with a Reactive Routing Protocol: AODV

The functionality of the AODV protocol [34] is based on maintaining a vector of paths (i.e. routing table) that lead to the different destinations at each node. A given node does not have full knowledge of any of the routes. It only knows the next hop along any given route. Each node keeps only one route (the one that has the smallest number of hops i.e. shortest route) to any given destination. AODV also uses periodic HELLO messages to keep neighbors aware of the other nodes in the neighborhood. This process is used for route maintenance. Our goal is to assess the effect of introducing PIES with its periodic node sleep states on this process.

For this purpose, we modified AODV so that it can function in cooperation with PIES. The main modifications included:

- Interaction between AODV and PIES to determine node sleep states and hence forward delays
- Inclusion of the determination of minimum threshold values on routing paths in AODV
- Inclusion of route selection based on minimum threshold values on routing paths
- Piggybacking of PIES info on AODV control packets

With a traffic data rate of 12kbps per sender, we ran the simulations for different pause times with AODV active alone and then with PIES enabled together with AODV.

With PIES enabled, and for the selected values for the mandatory sleep and wakeup times as indicated above, the energy performance was better than in the case of AODV alone as shown in Figure 5-1. We were able to reach energy savings close to 50% over the routing algorithm alone. The degree of improvement varied with the values of the ST and WT times used. In this figure, we notice that PIES energy consumption reduction for a given ST/WT pair value is consistent regardless of the mobility conditions of the network nodes. The figure also shows that PIES energy saving performance increases with the increase of the ST:WT ratio. When considering the 95% confidence intervals in the case where ST/WT = 0.75/1.0 with pause time = 600 sec as an example, we find that the energy savings are between 38.12% and 40.25%. This result shows that the energy savings that are achieved by PIES are statistically significant.

We also found that the standard deviation of node energies, which we use as an indicator for energy fairness between nodes, is better with PIES enabled. This is despite the fact that with PIES disabled, idle energy consumption presents an inherent source of energy balance between the nodes to some extent since wireless interfaces would consume energy even while in idle state. With PIES enabled, we were able to reduce idle energy consumption significantly and at the same time achieve balanced energy consumption

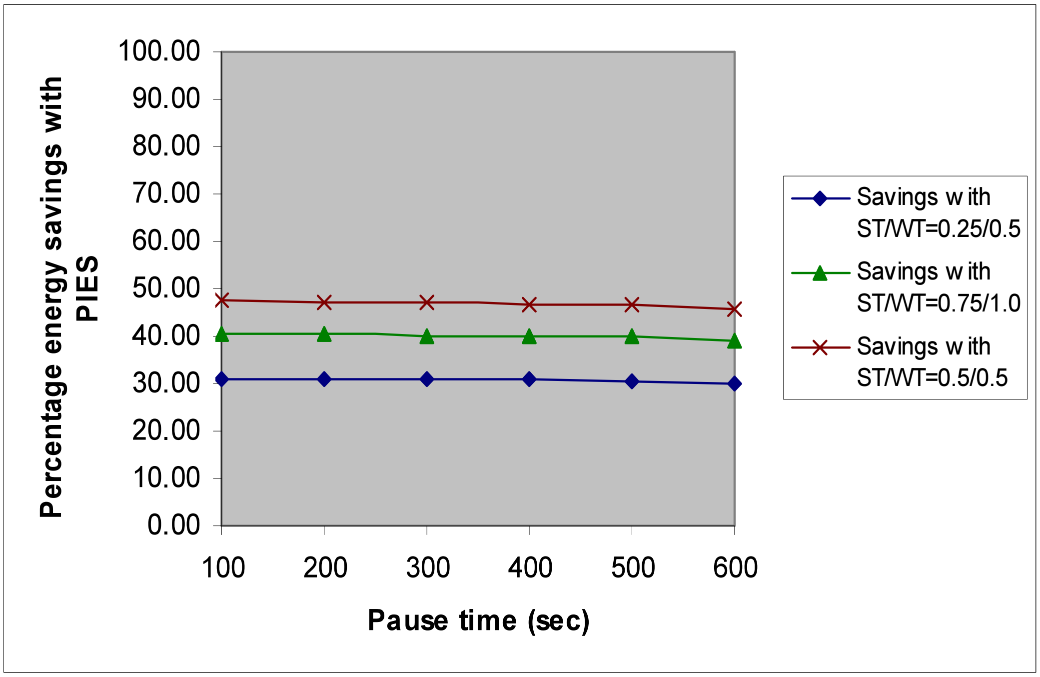


Figure 5-1: Energy savings with PIES enabled with AODV

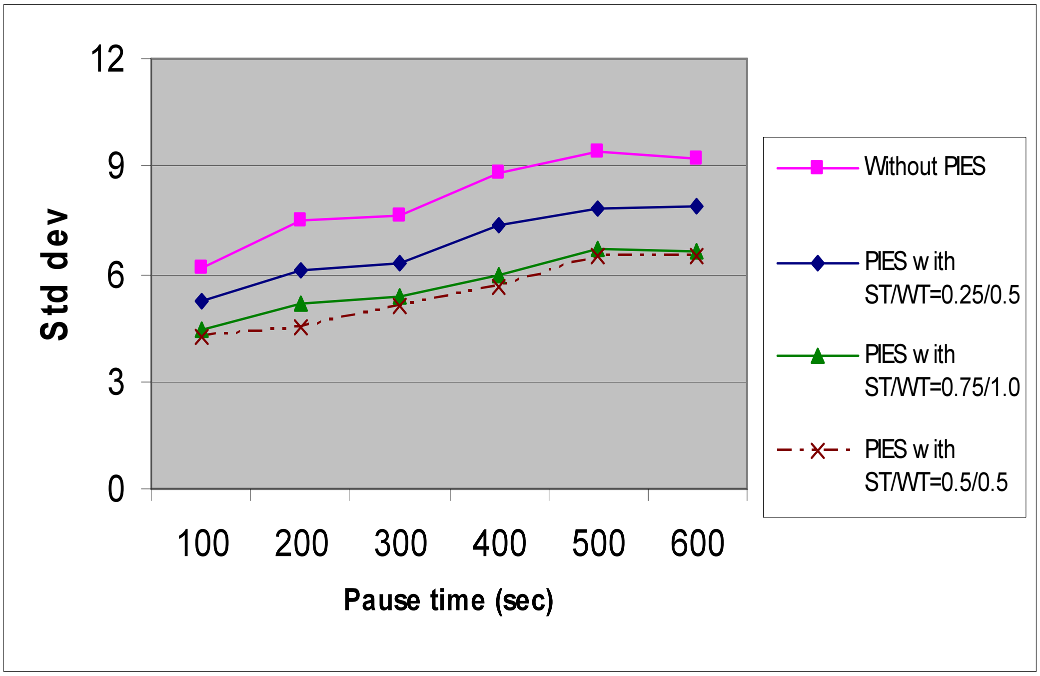


Figure 5-2: Standard deviations of node energies for AODV with and without PIES

between network nodes, exceeding the case when PIES is disabled. Figure 5-2 compares the standard deviation of node energies without PIES and with PIES enabled. We notice from the figure that the standard deviation decreases (and hence energy fairness increases) with the increase of the ST:WT ratio. We notice that energy fairness generally increases with increased mobility, as more nodes share more equally in the routing duty. Considering the 95% confidence intervals with ST/WT values of 0.75/1.0 and pause time = 600 sec, we found that without PIES enabled, the range of the standard deviation of node energies is [8.03, 10.41]. When PIES is enabled, the standard deviation range is [5.98, 7.31]. As we can see, there is no overlap between the two ranges which means that the difference in energy fairness between the two cases is statistically significant.

We have also measured the packet delivery ratios without PIES and then with PIES enabled. We found that, with PIES enabled and for the ST/WT pair values that we used, the impact on the packet delivery ratio (PDR) is not significant especially as the pause time increases (lower mobility). It also tends to drop slightly with the increase of the ST:WT ratio. This is clearer in the case where ST/WT=0.5/0.5 over the two cases where ST < WT in which PDRs are closer to each other, see Figure 5-3. We notice from this figure that packet delivery ratios become almost the same in the case where nodes are static both in case of AODV alone and in all the PIES cases.

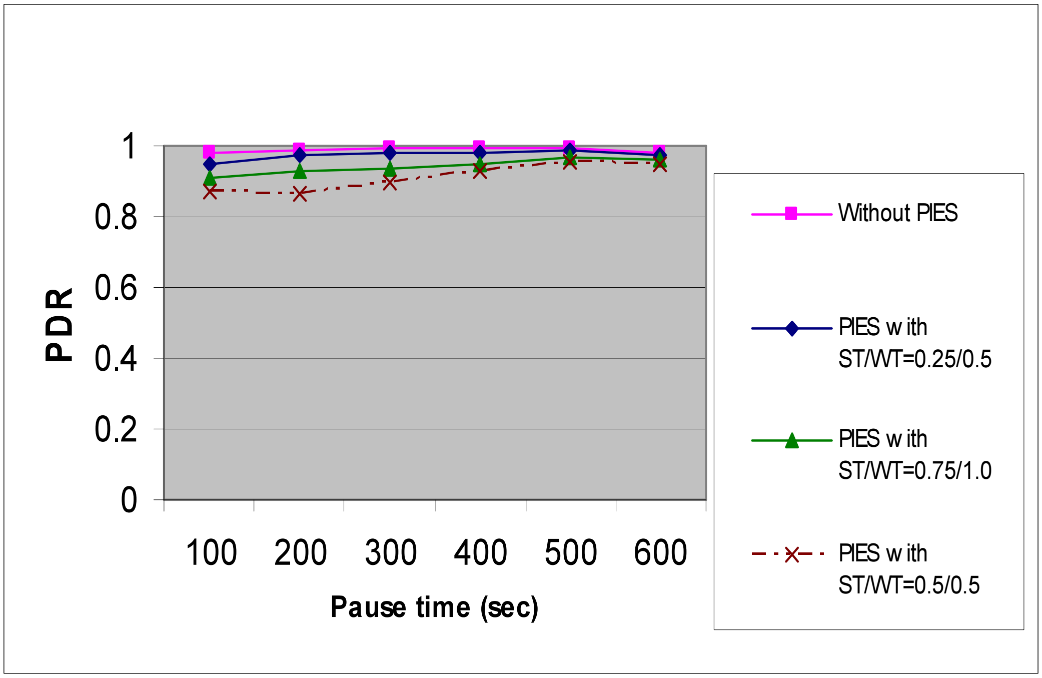


Figure 5-3: Packet delivery ratio for AODV with and without PIES

We also measured one other aspect of the performance which is packet delivery latencies. Here we found that PIES performs worse to varying degrees than when PIES is disabled (AODV alone). This is due to the fact that packets may have to wait before they get transmitted over several hops due to the next-hop neighbor being asleep. One possible way to reduce this impact is by the proper choice of the ST and WT parameters of PIES as we will discuss later. Figure 5-4 gives the packet latency comparison between the cases where PIES is enabled and without PIES. From this experiment, we see that a higher ST:WT ratio results in a better energy performance with no significant impact on packet delivery ratio but with increased packet delivery latency, for the values we used. We will be discussing the PIES additional latency analytically in Chapter 6.

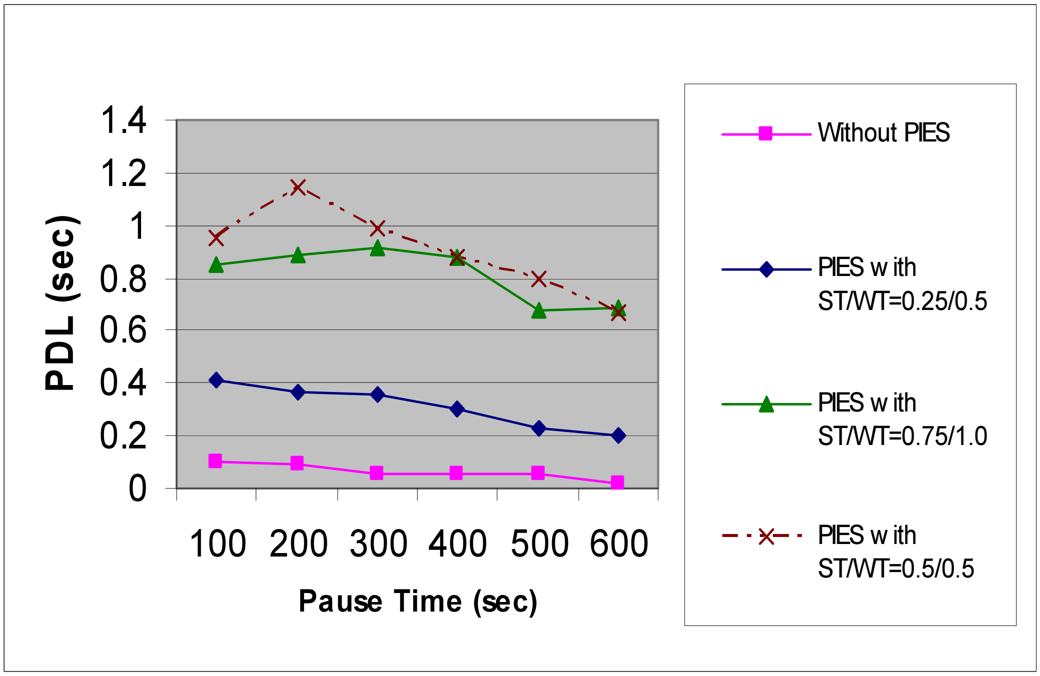


Figure 5-4: **Packet delivery latency for AODV with and without PIES**

5.4.2 Evaluation Results with a Proactive Routing Protocol: OLSR

Through Optimized Link State Routing (OLSR) [11], [21],[29], nodes exchange topology information on a regular basis. Selected multipoint relay (MPR) nodes announce this information periodically to the network. These nodes are also used in calculating the route from any network node to a given destination. MPR node selection depends on periodical exchange of HELLO messages between nodes. Also, topology control messages are sent periodically (or upon MPR node changes) by nodes to maintain topology information throughout the network. PIES operation relies on periodical sleep operations. Therefore, it is important to investigate its effect and adaptability to the operation of OLSR in which periodical message exchange is crucial to the health of the network.

To test the performance of the PIES algorithm with OLSR, we used a traffic data rate of 3kbps. This is lower than what we used with AODV to avoid congestion which results from the combination of the proactive OLSR route maintenance control messages and high traffic rates. In order to enable OLSR to function with PIES, we modified OLSR in a way similar to our modifications to AODV. In addition, we adapted the frequency of sending the OLSR control packets that are crucial to the health of route maintenance to the sleep cycle of PIES. This is done by resending copies of the control messages after ST seconds. We ran OLSR with PIES disabled and then with PIES enabled to compare the performance in both cases. In general, we found the results to have similar trends as in the case of AODV with slight differences. The energy savings are slightly lower than the case of AODV, see Figure 5-5. The reason is the proactive nature of the OLSR protocol which requires the periodic exchange of topology maintenance messages which consumes more energy when nodes are awake than would be the case with a reactive routing protocol. When we consider the 95% confidence intervals of energy savings, and considering the case where ST/WT = 0.75/1.0 and pause time = 600 sec, we found that the range of energy savings is [36.28, 37.08] which is a tight range. As far as energy fairness is concerned, we found that PIES helps achieve considerably better results than in case of OLSR alone, see Figure 5-6. This can be attributed to the fact that relay nodes, which are the nodes that are used for routing duties in OLSR, are also given a fair share of sleep time and consequently energy savings when PIES is enabled. When we consider the 95% confidence intervals as above, we find in the case where PIES is disabled, the range of the standard deviation of node energies is [5.2, 5.63] while with PIES enabled it is [3.65, 4.13]. We can see that there is no overlap between the two ranges and the result is therefore statistically significant. As far as packet delivery performance is concerned, enabling PIES resulted in a comparable packet delivery ratio with the case where PIES is disabled especially with lower ST:WT ratios. As in the AODV case, the PDR tends to drop with the increase of the ST:WT ratio, see Figure 5-7. As for packet delivery latency, we found that it increases with the increase of the sleep time, ST, as shown in Figure 5-8.

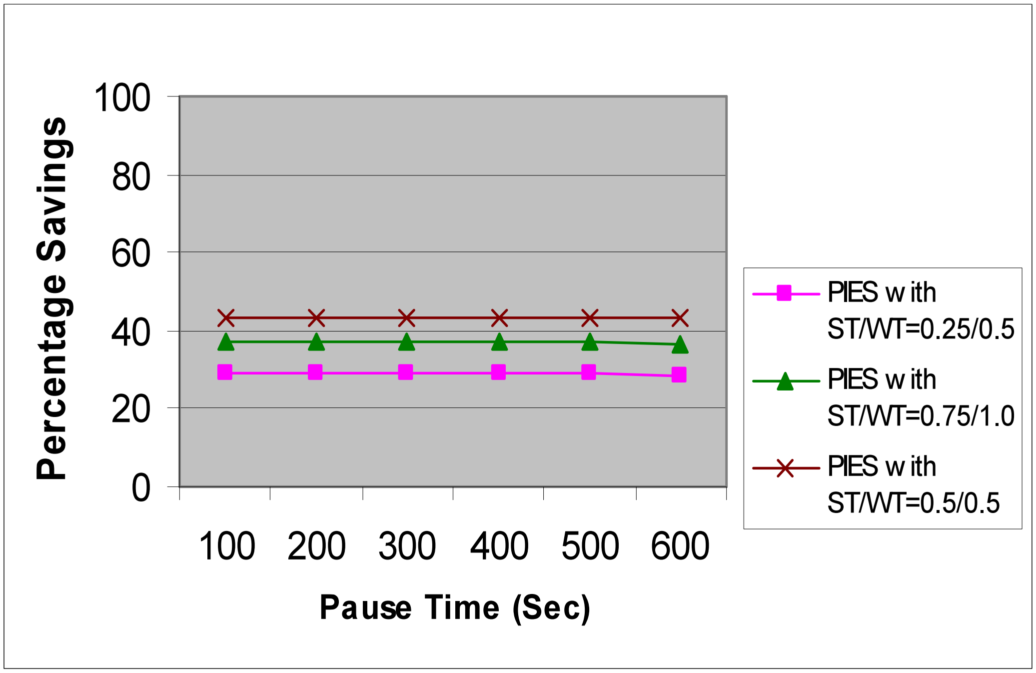


Figure 5-5: Energy savings with PIES enabled with OLSR

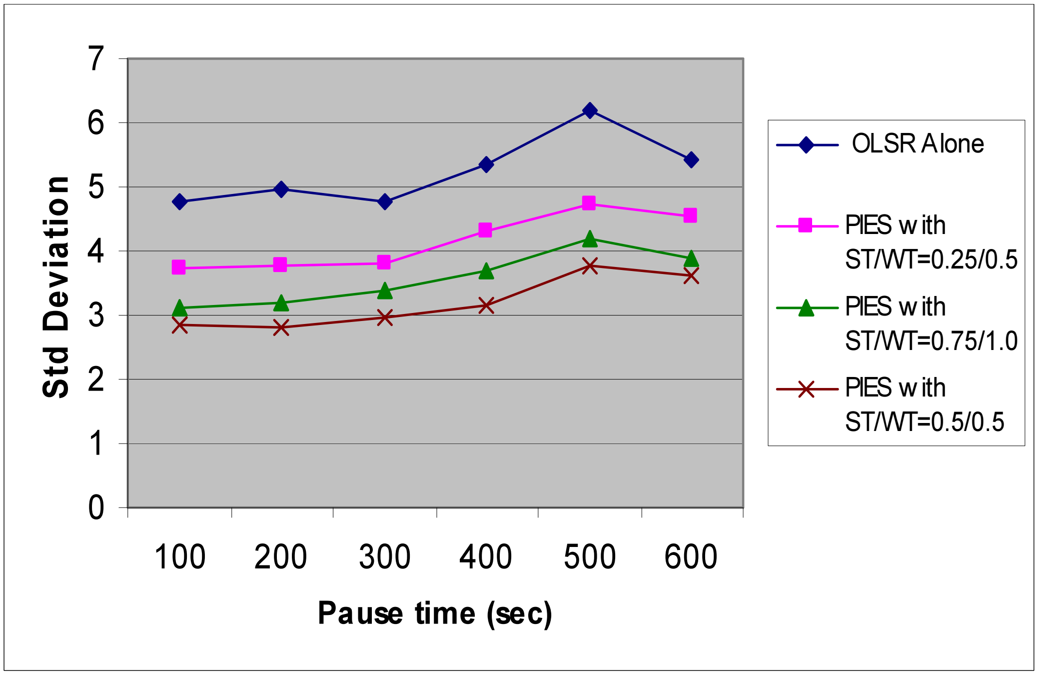


Figure 5-6: Standard deviations of node energies for OLSR with and without PIES

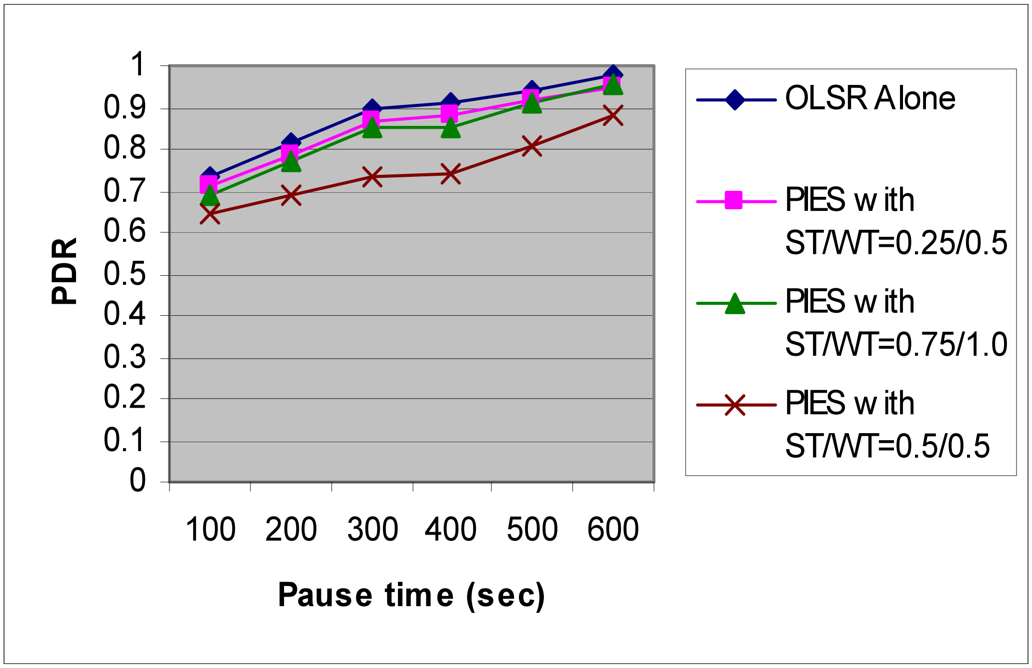


Figure 5-7: Packet delivery ratio for OLSR with and without PIES

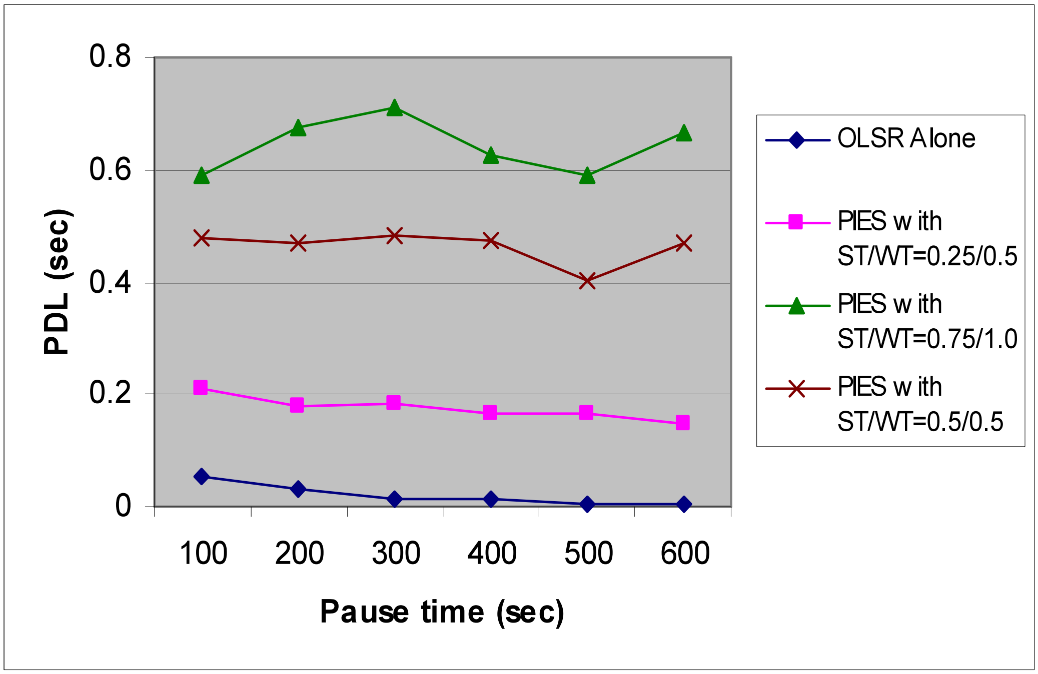


Figure 5-8: Packet delivery latency for OLSR with and without PIES

5.5 Scalability of the PIES Algorithm

In this section, we examine the ability of the PIES algorithm to scale with the increase of the network population as well as with the increase of traffic data rate. The goal here is to measure the ability of PIES to continue to deliver operational performance that is comparable to that of the routing protocol alone, and energy performance that is superior to that of the routing protocol alone for bigger networks.

5.5.1 PIES Scalability Performance with Higher Network Population

We performed the experiment of Section 5.4.1 with networks that have 75 and 100 network nodes, respectively. We used the mobility scenarios based on this number of nodes, and the rest of the simulation conditions are kept the same. We used AODV as the routing protocol. For PIES, we used ST/WT values of 0.25/0.5.

The increase of network nodes increases possibility of collisions with each network node trying to advertise its presence according to the routing protocol strategy, in this case AODV HELLO messages. Also, the possibility of more routes and route replies to requests increases, hence increasing the traffic load on the network.

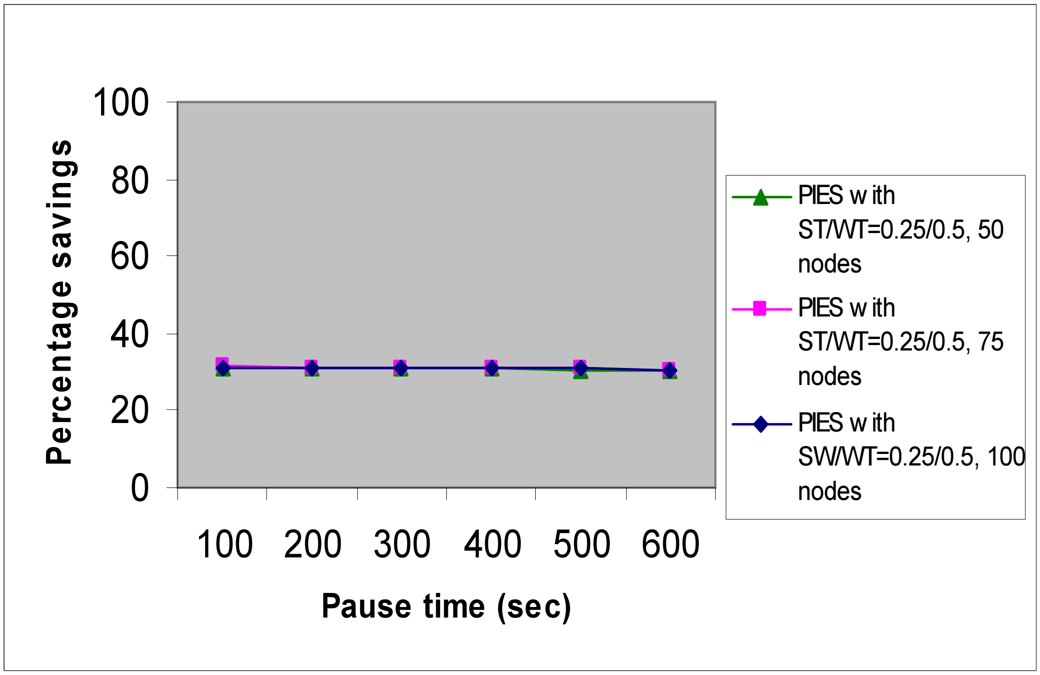


Figure 5-9: Energy savings with PIES enabled with AODV in a network with 50, 75 and 100 nodes with ST/WT = 0.25/0.5

As far as energy performance is concerned, we see from Figure 5-9 that PIES in this case resulted in around 30% energy savings over AODV alone for this ST/WT pair for both cases. This savings level is almost the same as what we saw previously in the case of 50 nodes. We have included the trend for the 50-node network case in this figure for easy comparison.

The standard deviation trend shown in Figure 5-10 is also more or less the same as in the case of 50 nodes, for both the 75 and 100 node cases. Looking at the network operation performance, we see that the packet delivery ratios are comparable in the case when PIES is enabled to that where AODV is functioning alone, see Figure 5-11. We notice a slightly better PDR performance in general in all cases than in the case with 50 nodes due to the fact that more nodes are available which results in more routes to destinations due to the denser nature of the network. By examining the packet delivery latency trend in the case where PIES is enabled and where PIES is disabled, we find that the trend is again similar to that in the 50 nodes case but with a narrower latencies difference between the case of PIES and that of AODV alone. We attribute this again to the denser network that enables shorter routes with more possibilities of nodes awake on these routes which can reduce the time a route is discovered and used.

From this experiment, we see that PIES continues to perform well with higher network node population. In fact, it shows some improvement with regard to the packet delivery latency as we explained above.

5.5.2 Scalability of PIES with Different Traffic Rates

In this section, we measure the performance of PIES as the network traffic increases. In order to perform this evaluation, we use the same network conditions as indicated previously but with several data rates. At each of the chosen send rates, we measure both the energy performance and network operation performance and compare the case when

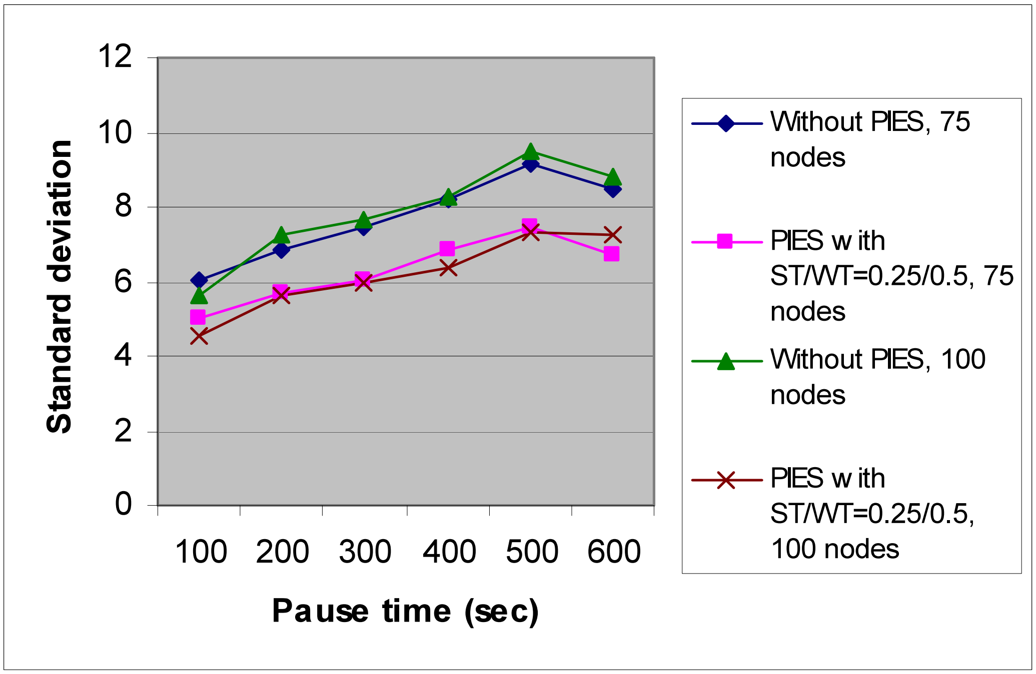


Figure 5-10: Standard deviations of nodes energies with AODV in networks with 75 and 100 nodes

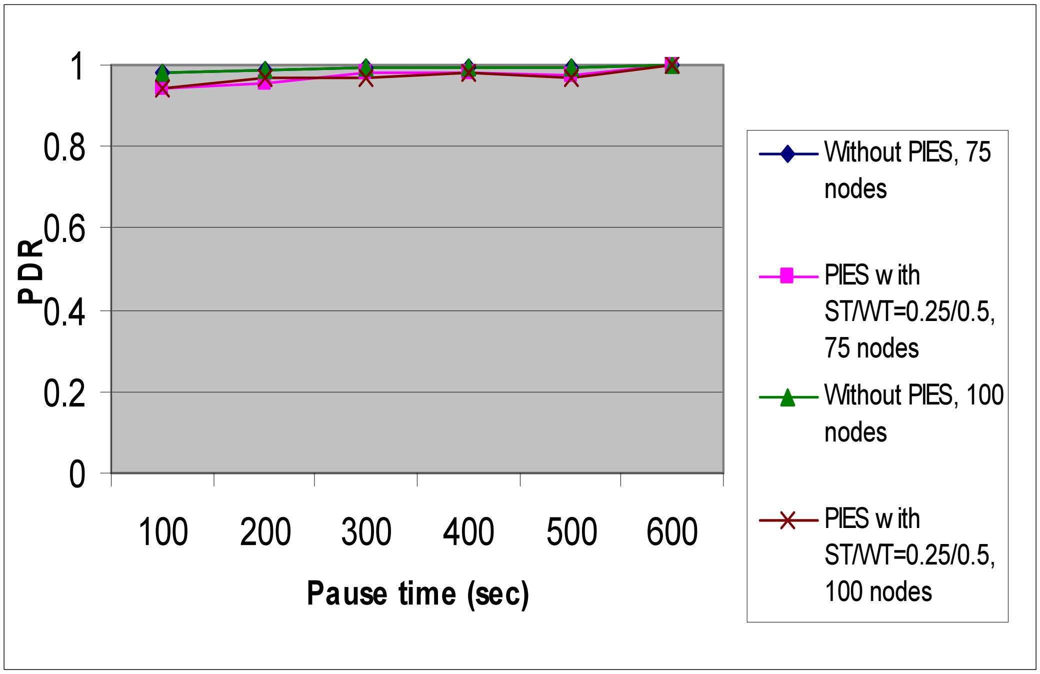


Figure 5-11: Packet delivery ratios with AODV in networks with 75 and 100 nodes

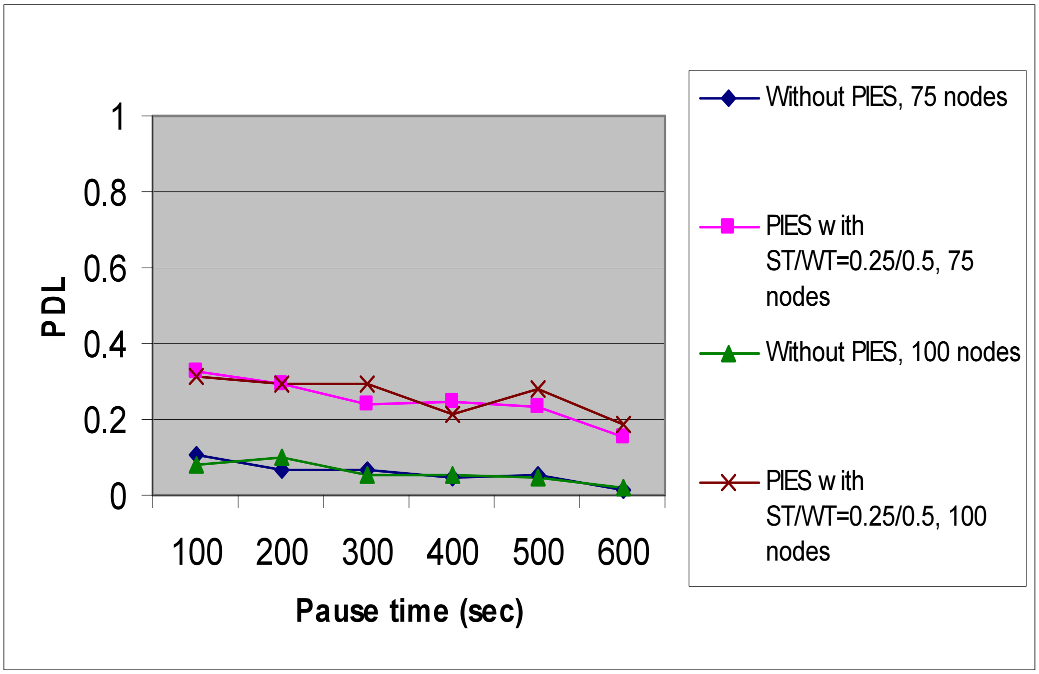


Figure 5-12: Packet delivery latencies with AODV in networks with 75 and 100 nodes

PIES is enabled with that when it is not. We used AODV as the routing protocol and ran the simulations for the case where pause time = 600 seconds, i.e. with no motion.

From Figure 5-13, we see that the energy savings of PIES over AODV alone is consistent regardless of the traffic send rate. This is also generally the case as far as energy fairness is concerned which is evident from the trend shown by Figure 5-14.

Considering network operation performance, Figures 5-15 and 5-16 show the packet delivery ratios and packet delivery latencies respectively with the different traffic send rates. From these figures, we see that the packet delivery ratio performances when PIES enabled and when it is disabled are identical. At a very high rate, 15 Kbps, we start seeing a slight drop when PIES is enabled. This drop is due to interface queue overflow that occurs as the data rate increases together with the fact that PIES causes packet buffering while nodes are asleep. It is worth mentioning that when we increased queue sizes, we were able to lower this drop which is already very small. We have not reflected this in the results, however, since this increase in the queue size is not needed in the case of the routing protocol alone. As for the packet delivery latencies, we see that the difference is consistent for low, medium and relatively high data send rates. As the traffic data rates get higher, we see that the latencies with PIES enabled start to increase at a higher rate. This can be attributed to increased contention at the communication channel due to the buffered traffic that results from nodes having been in sleep state combined with the already constantly high traffic rate.

From this experiment, we see that the PIES performance is consistent with the higher network traffic both from energy and network operation points of view. As the traffic becomes quite high, the PIES-induced latency increases further.

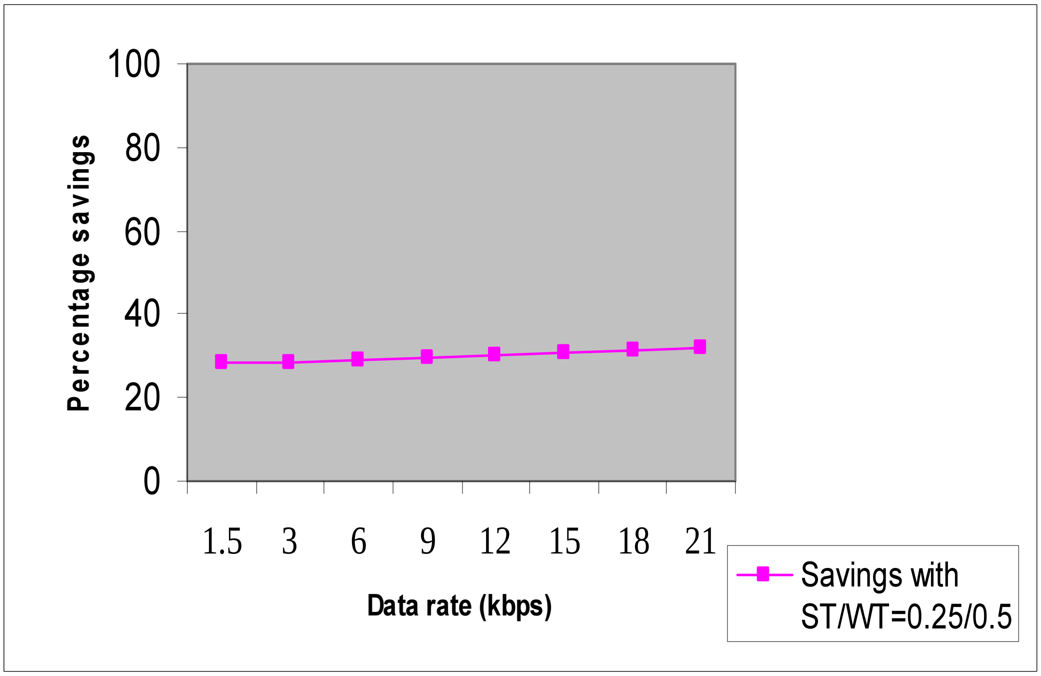


Figure 5-13: PIES energy savings over AODV alone with different traffic rates

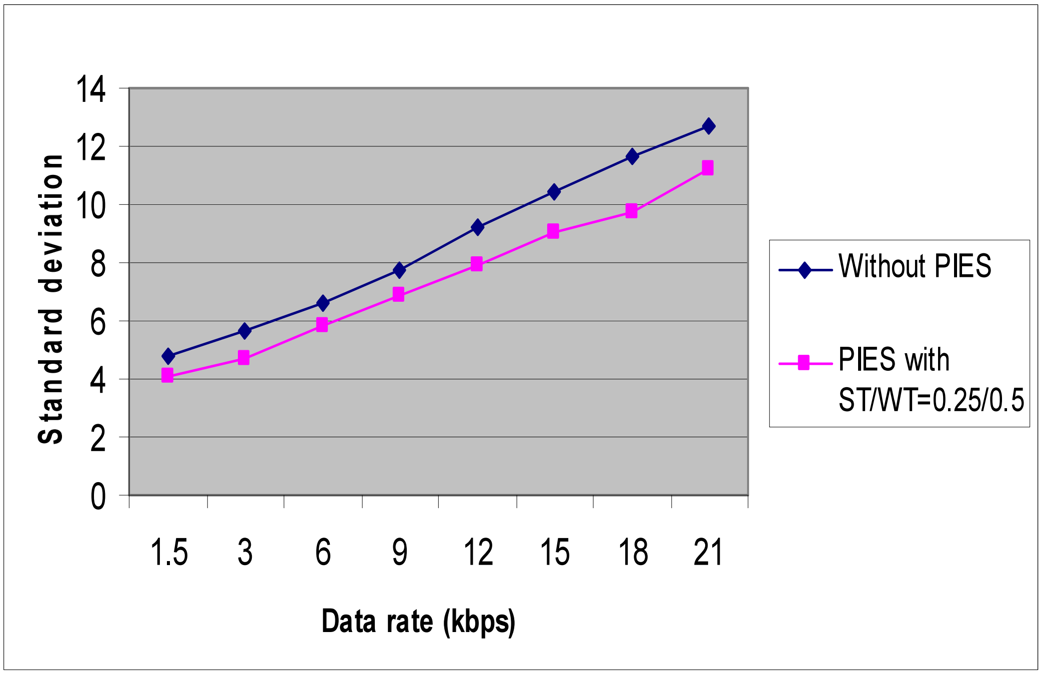


Figure 5-14: Standard deviation of node energies with different traffic rates

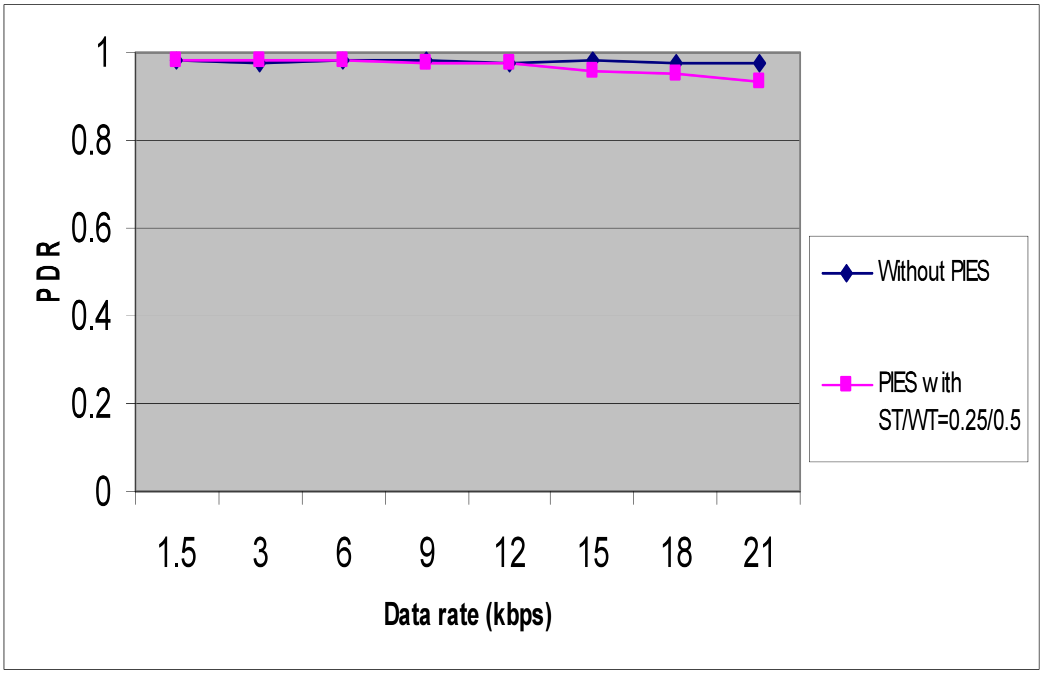


Figure 5-15: Packet delivery ratios with different traffic rates

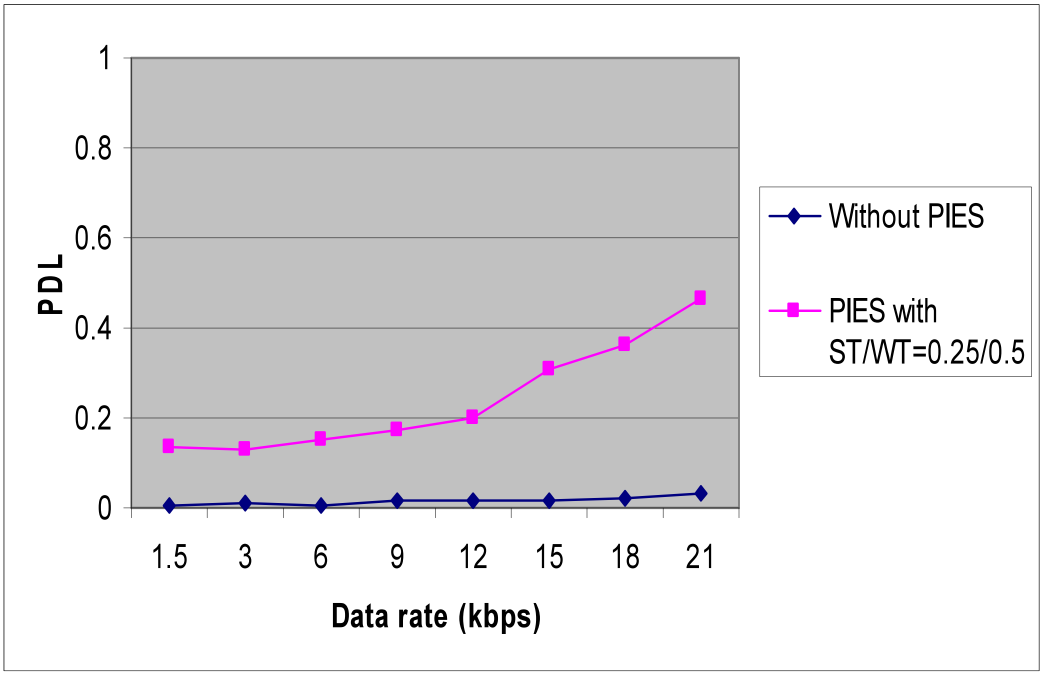


Figure 5-16: Packet delivery latencies with different traffic rates

5.6 PIES Performance with On-Off Traffic Sources

On-off traffic represents a bursty traffic model that is frequently used to characterize traffic in data networks. In this model, traffic sources switch between on and off traffic sending states. They remain in each state for a certain period of time, namely the busy and idle periods, respectively. This type of traffic can result in a higher rate of collision, delays and network congestion [31],[36]. We found it important to study the effect of enabling PIES under these traffic conditions to measure its ability to cope with harsher network conditions.

We have experimented with this traffic model with AODV as the routing protocol. The busy and idle intervals of the traffic follow exponential distributions with means of 10 and 100 seconds respectively. In this experiment, we used 20 traffic connections with 3 Kbps data rate for each traffic source during busy intervals.

If we consider the energy performance, we find that PIES provides consistent energy savings over AODV alone regardless of the mobility conditions of network nodes, see Figure 5-17. Also, PIES still shows a higher degree of fairness over AODV alone even in this bursty condition which implies higher degree of node utilization in general to establish routes and recover from collisions. Figure 5-18 shows this trend. It is worth mentioning that the standard deviation trends as shown in Figure 5-18 are generally lower than the cases with CBR traffic (e.g. as in Section 5.4.1). This is due to the fact that node utilization is generally higher across network nodes in the case of bursty traffic than in the case of CBR traffic.

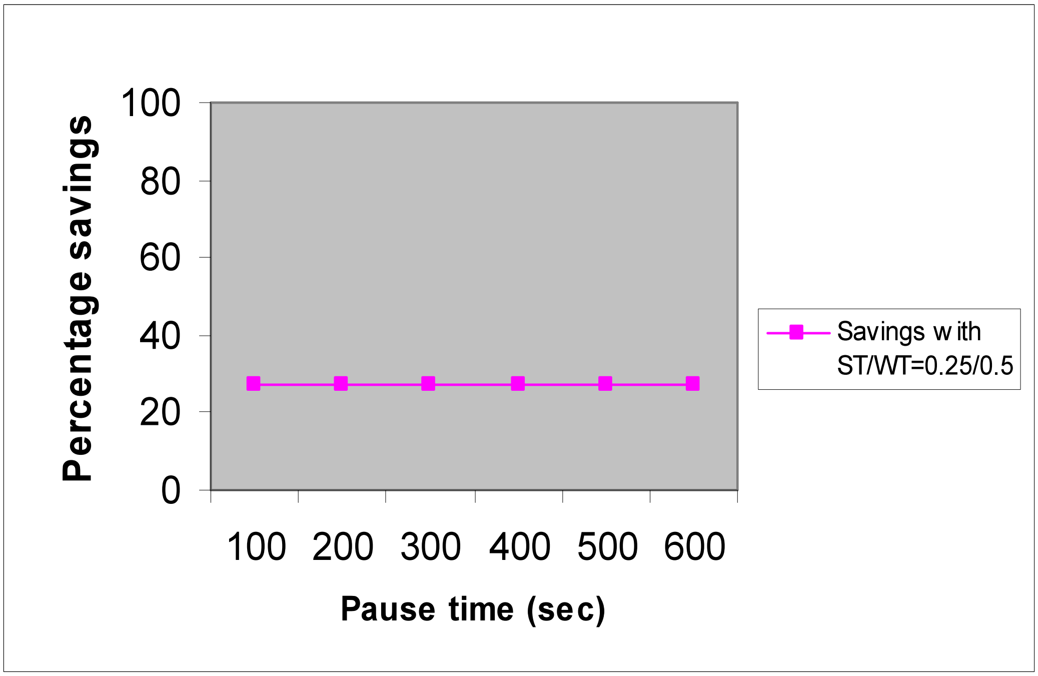


Figure 5-17: PIES energy savings over AODV alone with on-off traffic

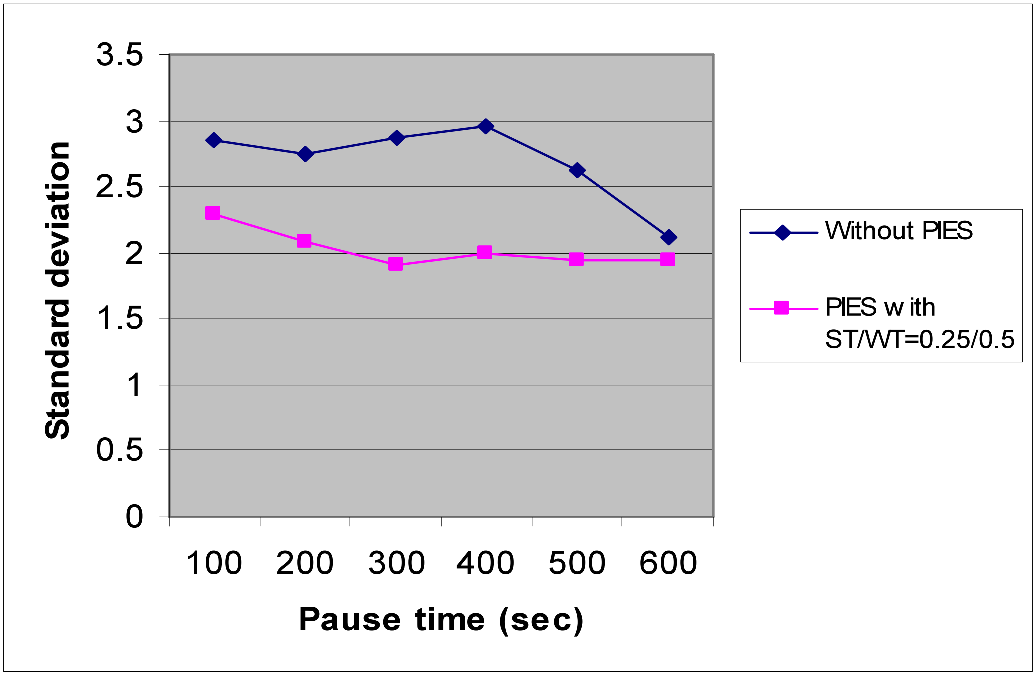


Figure 5-18: Standard deviation of node energies with on-off traffic

Considering network operation performance, we found that the PDR with PIES enabled is comparable to that when AODV active alone, see Figure 5-19. With AODV alone, packet delivery latency seems to be high with higher mobility conditions due to collisions. This narrows the difference between this case and the case when PIES is enabled considerably, since PIES mandates that nodes sleep periodically and hence reduces contention. Figure 5-20 shows the packet delivery latency behavior.

From this experiment, we see that PIES continues to show good energy performance that is similar to that of the case of long-lived traffic connections. As for network operation performance, PIES seems to adapt well to the conditions imposed by bursty traffic and seems to behave even better than in the case of the long-lived traffic connections relative to AODV alone especially under higher mobility conditions.

5.7 Effect of Changing PIES Parameters on the Performance

PIES has three important parameters: mandatory sleep time (ST), mandatory wakeup time (WT) and sleep separation factor. We have already seen the effect of changing the ST/WT pairs while we were experimenting with the different kinds of routing protocols earlier in this chapter. However, from this perspective, there is still an interesting case to be investigated: fixing the ST:WT ratio while changing the absolute values. This can give us a good insight into the relationship between the ST:WT ratio and the network operation performance as well as the energy performance. We also need to investigate the effect of changing the sleep separation factor on performance.

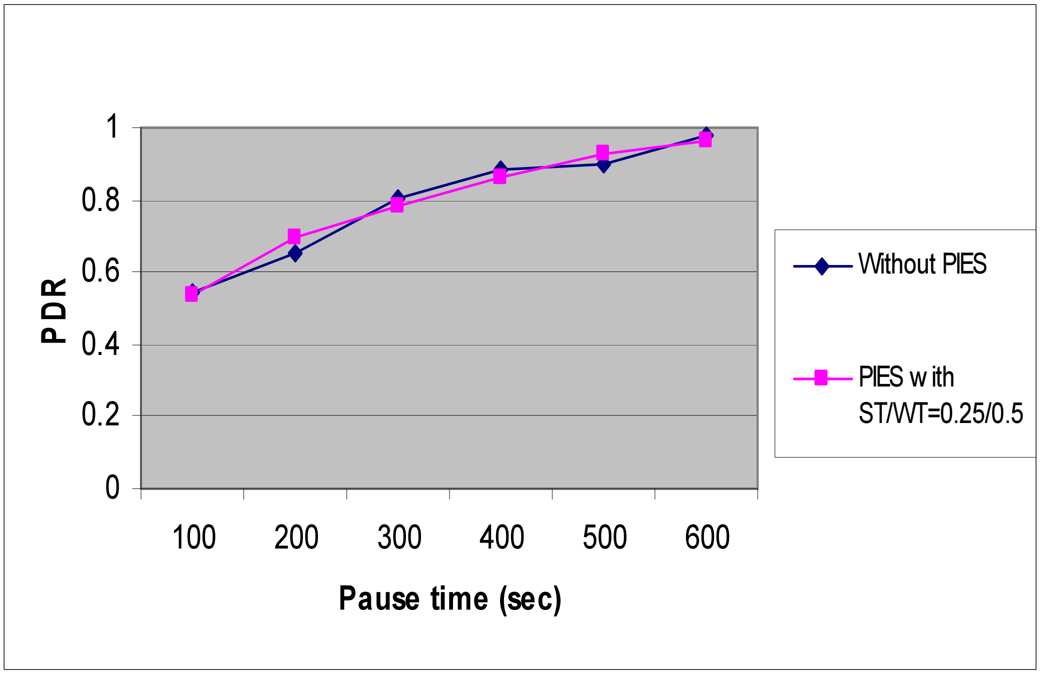


Figure 5-19: Packet delivery ratios with on-off traffic

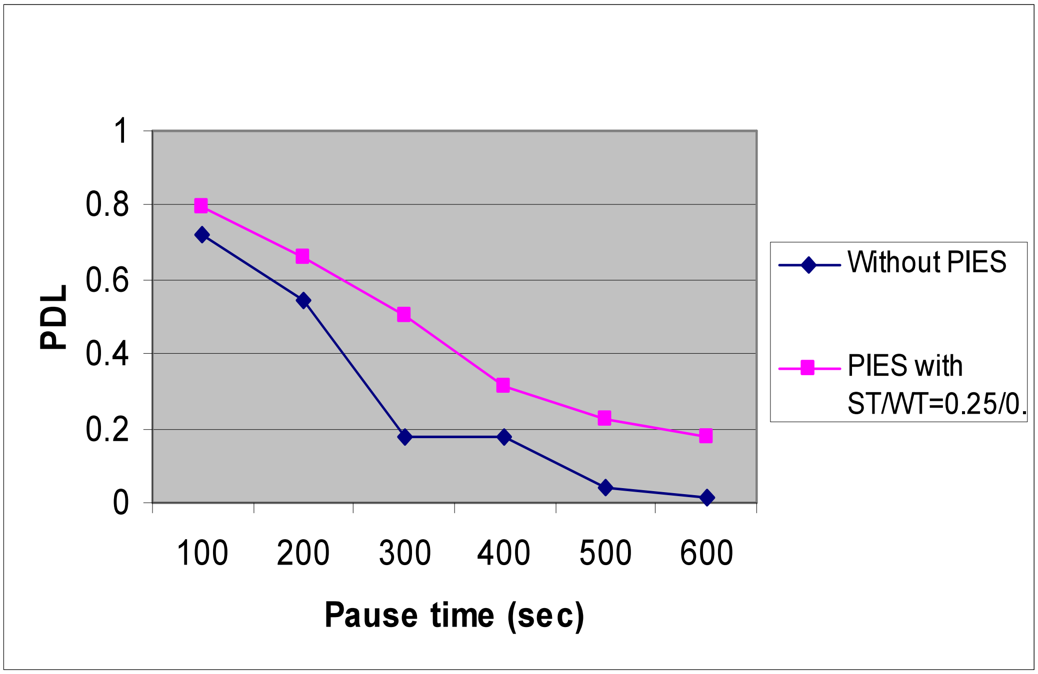


Figure 5-20: Packet delivery latencies with on-off traffic

5.7.1 Changing ST/WT Pair while keeping the same ST:WT Ratio

In this experiment, we ran simulations with AODV as the routing protocol for a pause time of 400 sec and used the following ST/WT combinations: 0.25/0.5, 0.5/1.0, 1.0/2.0, 1.5/3.0, and 2.0/4.0. The data rate per connection is 12 Kbps. We found that energy savings over AODV alone are almost the same in all cases, as shown in Figure 5-21. The standard deviation of remaining energies generally decreases as the values of ST and WT increase. This can be attributed to higher utilization of the wakeup period in handling traffic while addressing idle energy more aggressively with longer sleep times. This situation results in forcing a more fair distribution of routing duties at this mobility condition as the difference in node energies become more visible between nodes that just woke up and others that have been awake over the past period of time. Figure 5-22 shows this trend. As far as packet delivery ratio and packet delivery latency are concerned, the use of small values for ST and WT yields the best results for the load conditions that we used. The ST/WT = 0.25/0.5 in our experiment gave the best results for these metrics. For these network conditions, as the sleep time (ST) increases, the number of buffered packets increases which increases the chance of collisions and hence packet loss. The relatively short wakeup time does not affect route establishment as the load conditions leave reasonable room for exchanging route information within this time. Figure 5-23 shows the packet delivery ratio behavior. The latency also increases with the increase of sleep time of nodes along the path between the source and destination. Figure 5-24 shows the packet latency trend.

This experiment gives us some good insights into the choice of the ST/WT values. It shows that energy performance generally remains the same for the same ST:WT ratio. This is the case while the network operation performance is affected more by the value of ST in particular. As this value increases, the latency tends to increase and the PDR tends to drop. Therefore, we should choose relatively small values for ST while keeping the same ST:WT ratio in order to achieve a certain level of energy performance. This has the down side of achieving less fairness as shown in Figure 5-22. However, PIES in this case still achieves better fairness than in case of the routing protocol alone.

5.7.2 Effect of Changing the Sleep Separation Factor

As we mentioned in Chapter 4, it is important to ensure that not all nodes of the network have fully overlapping sleep periods. Therefore, we introduced what we call the *sleep separation factor* which determines the start of the initial sleep period of each node.

In this experiment, we need to determine the effect of choosing the value of the sleep separation factor on performance. For this purpose, we ran simulations with AODV as the routing protocol for static network conditions and used the ST/WT pair of value 0.25/0.5. Under these conditions, we changed the sleep separation factor to take the values 0.05ST, 0.1ST, 0.15ST and 0.2ST. The data rate per connection is 12 Kbps. From an energy performance perspective, we found that energy savings over AODV alone are not affected by the change of the value of the sleep separation factor, see Figure 5-25. The standard deviation of the remaining energies does not seem to be affected much either, see Figure 5-26. The packet delivery ratio and the packet delivery latency are

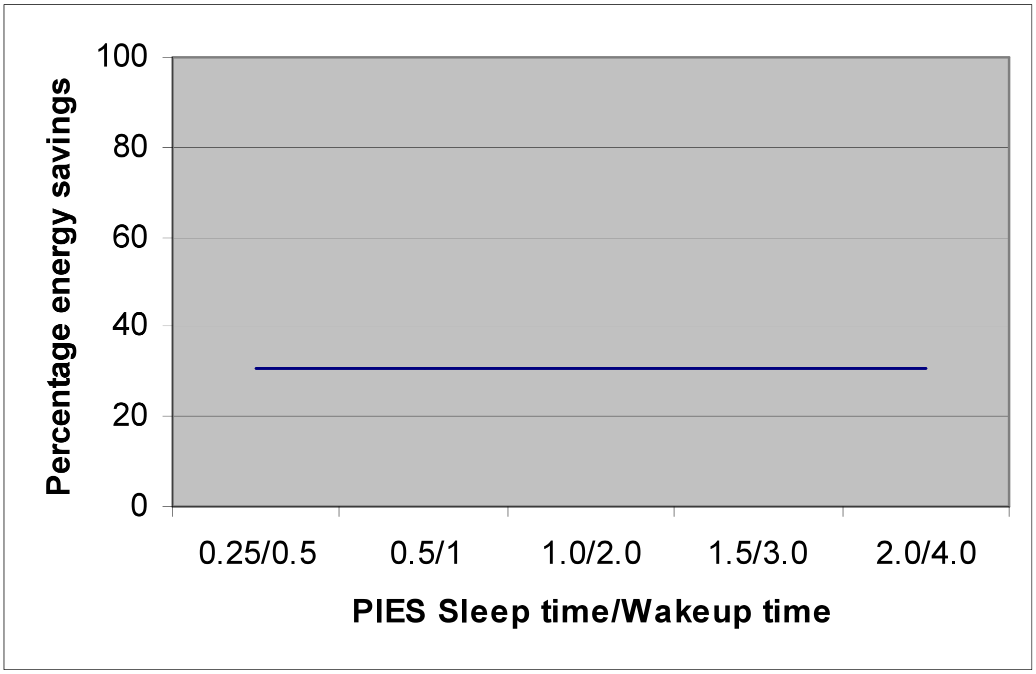


Figure 5-21: PIES energy savings for different values of ST and WT but with same ST:WT ratio

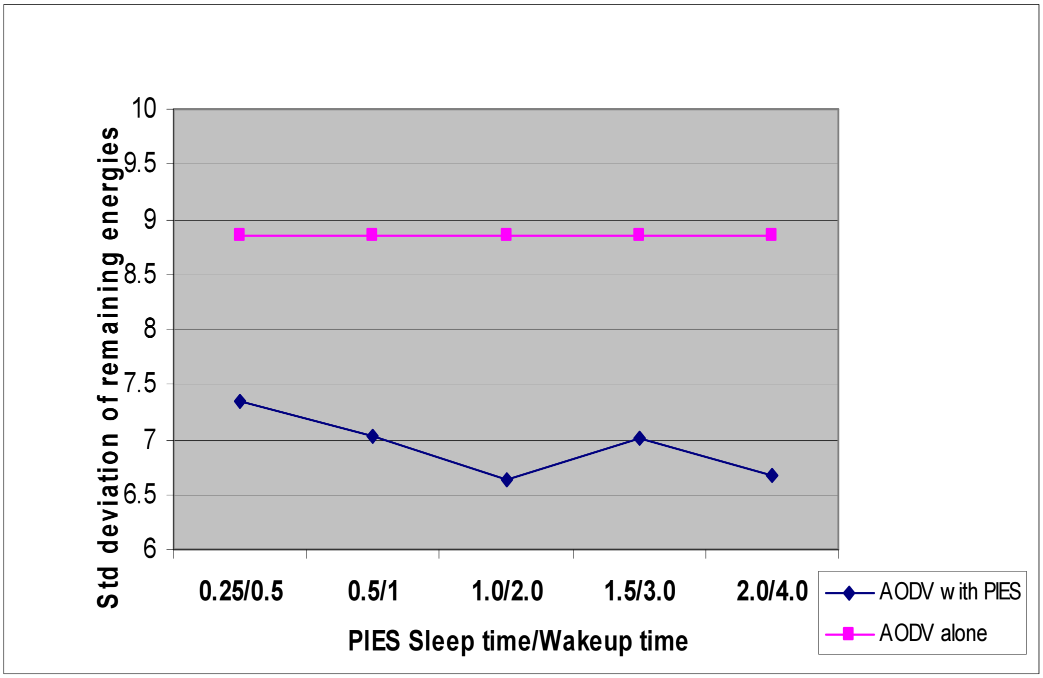


Figure 5-22: Standard deviations of remaining energy for different values of ST and WT (same ST:WT ratio)

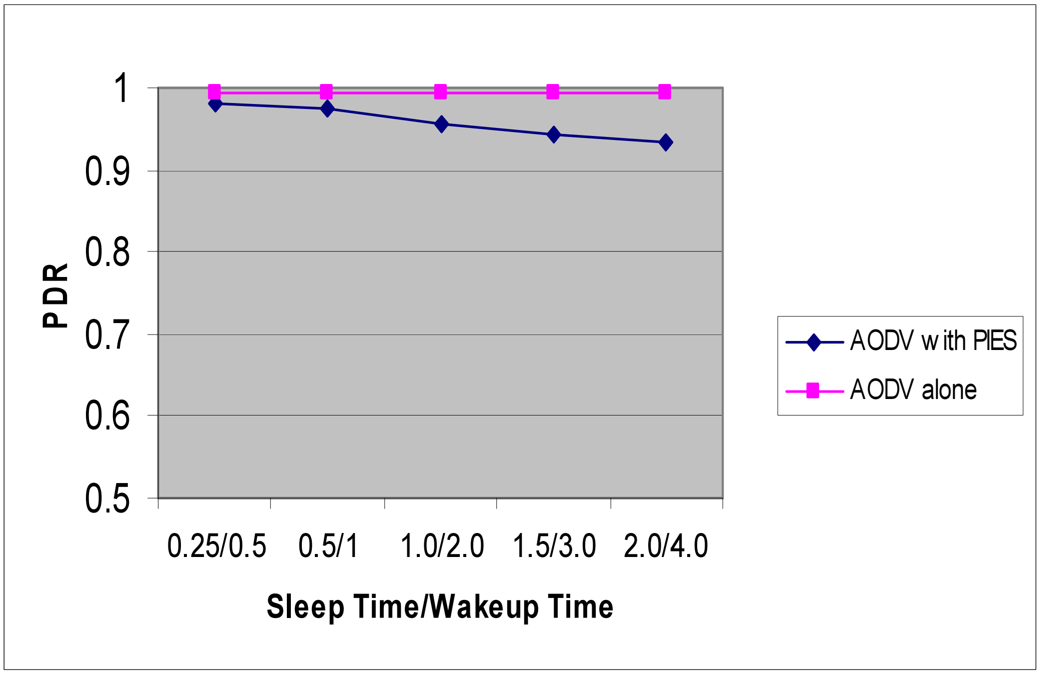


Figure 5-23: Packet delivery ratios for different values of ST and WT but with same ST:WT ratio

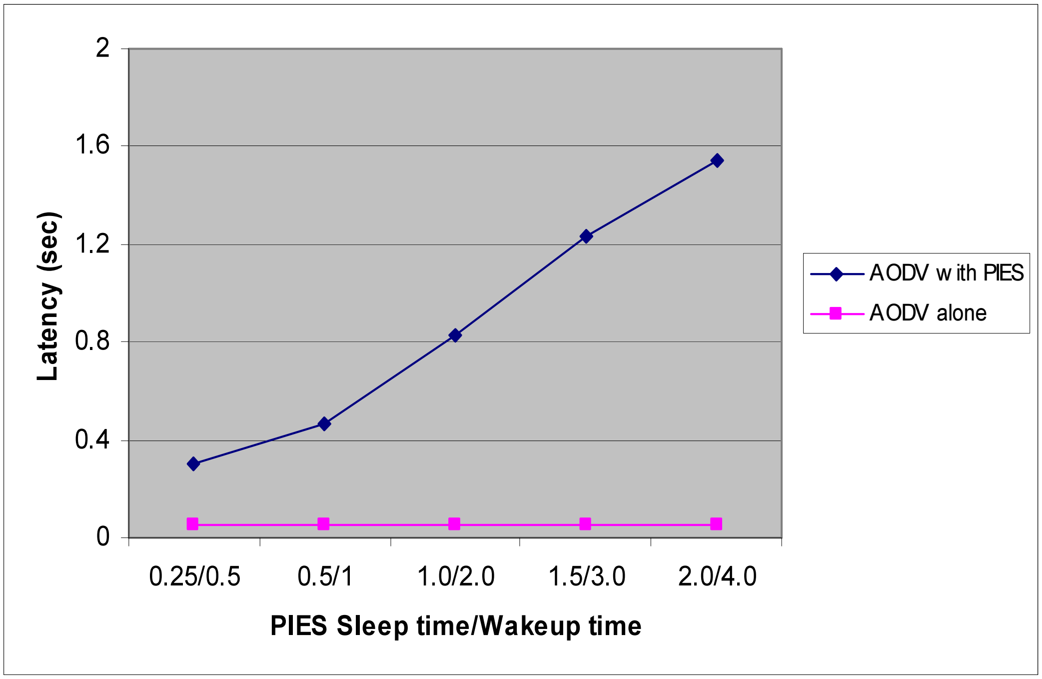


Figure 5-24: Packet delivery latency for different values of ST and WT but with same ST:WT ratio

almost unaffected by the value of the sleep separation factor, see Figures 5-27 and 5-28 respectively.

From this experiment, it seems that PIES performance is not sensitive to the choice of the value of the sleep separation factor. As long it is chosen to be of medium value (e.g. small fraction of the ST ~ 0.1-0.2 ST), there is no considerable resulting effect on performance. This is a welcome result since it makes it easier on the network operator to make a decision on the choice of the parameter without fearing negative impacts of their choice.

5.8 Effect of PIES on Network Lifetime

To examine the effect of PIES on extending the network lifetime, we ran simulations with AODV as the routing protocol with and without PIES enabled. We used a pause time value of 400 sec, initial node energies of 300 Joules and we ran the simulations for

600 seconds. The traffic rate is 12 Kbps per traffic source. For PIES, we use ST/WT=0.75/1.0. In the case when PIES is disabled, nodes start to die after 334.6 seconds of the simulation. At around 352 seconds, network operation comes to a complete stop with all the nodes vital to the operation failing. With PIES enabled, the first node failure occurred at around 514.6 seconds. Other nodes started to fail until the network operation ceased with the nodes vital to the network operation running out of energy around 542 seconds of the simulation. This presents a network lifetime extension of about 54% over AODV alone. This percentage extension in network lifetime is higher than the percentage energy savings that we obtained in earlier experiments. This is

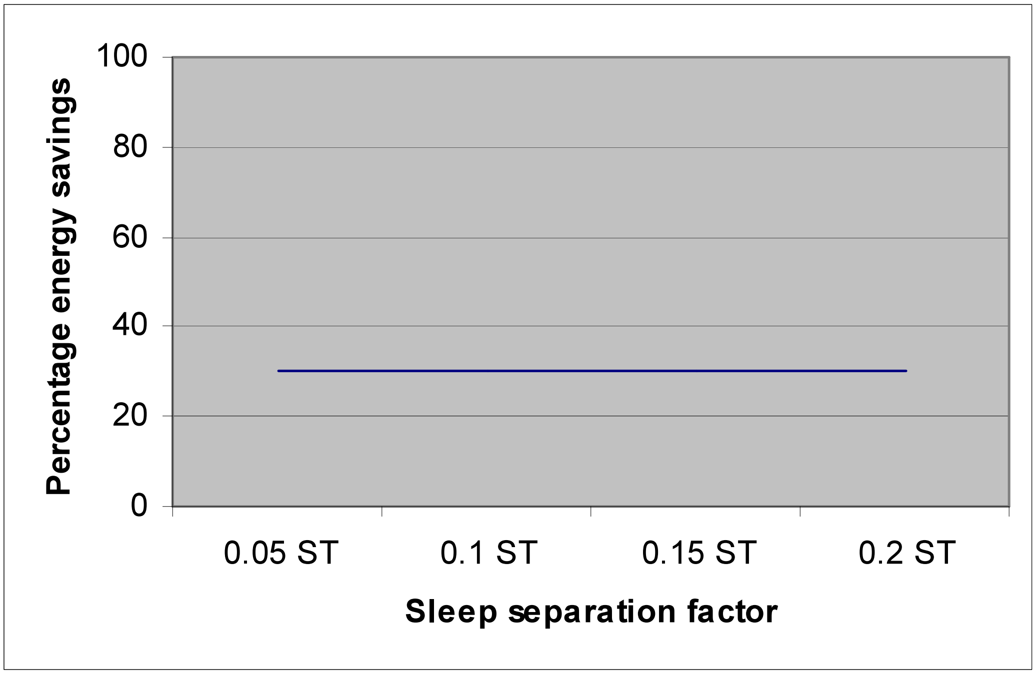


Figure 5-25: PIES energy savings for different values of sleep separation factor

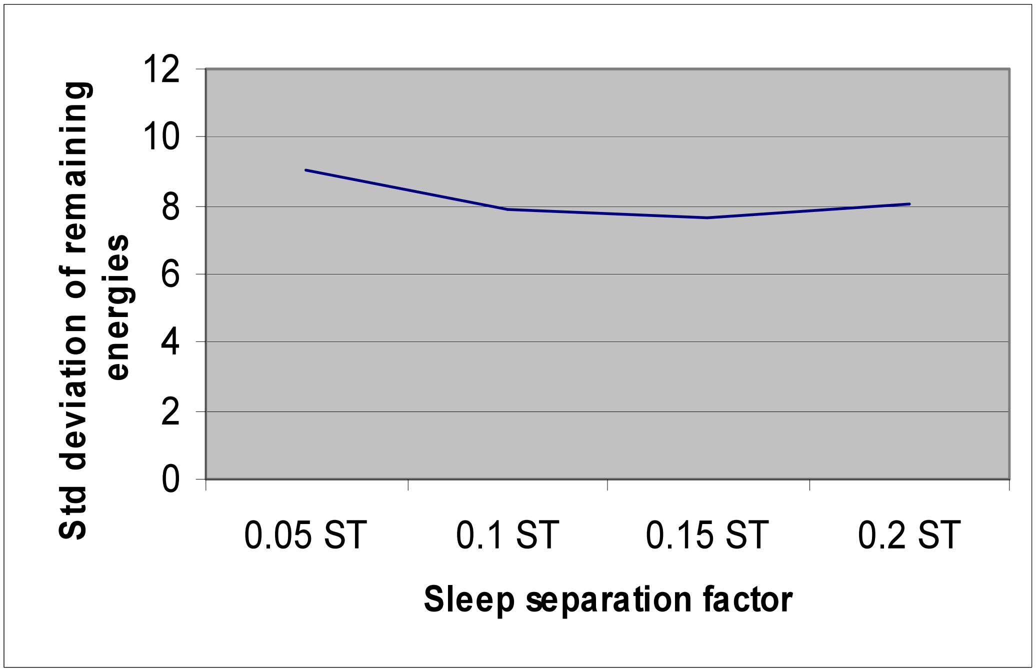


Figure 5-26: Standard deviation of node energies for different values of sleep separation factor

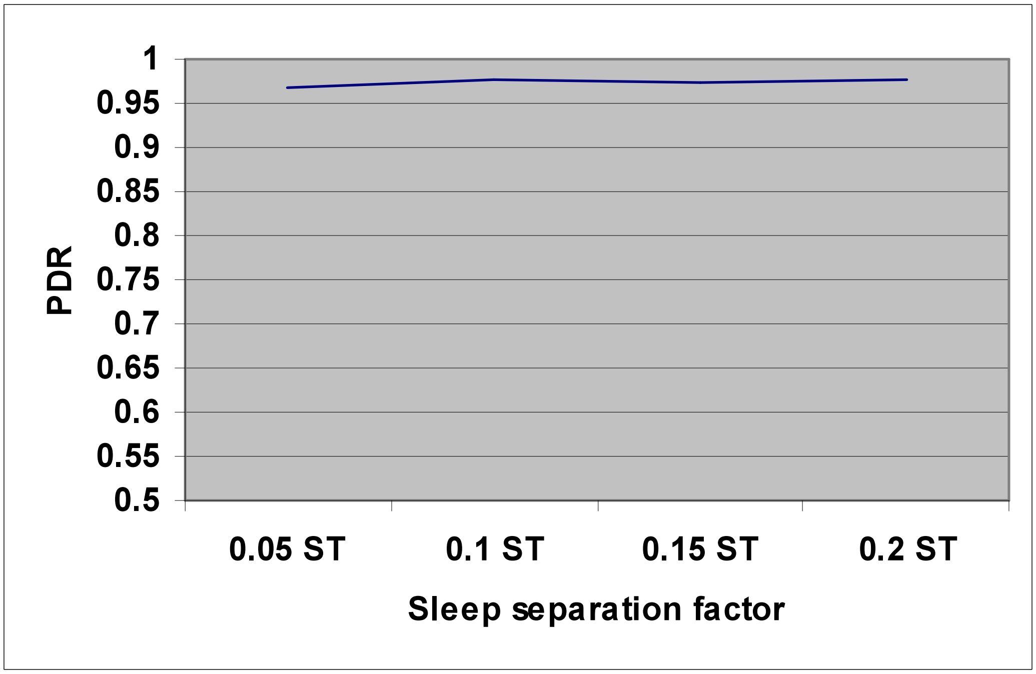


Figure 5-27: Packet delivery ratios for different values of sleep separation factor

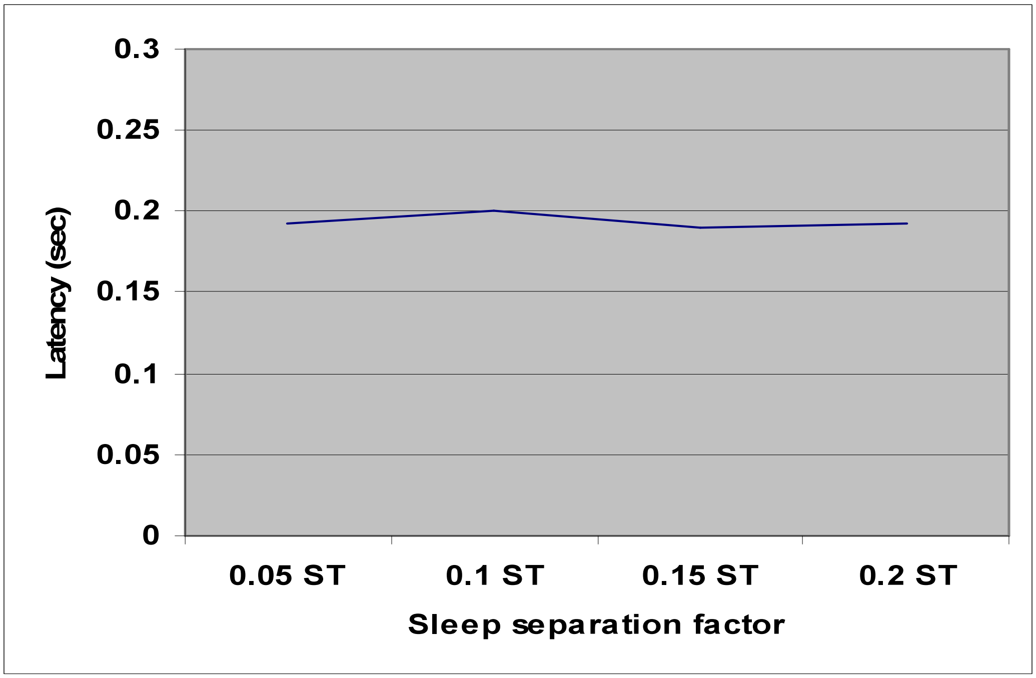


Figure 5-28: Packet delivery latencies for different values of sleep separation factor

because the energy savings are calculated as average across all nodes while the lifetime of the network may depend on some remaining nodes that possess energy higher than the average remaining network node energy. In addition, the higher imbalance of energy consumption in the case of AODV alone (without PIES) causes faster depletion of the energy of some critical nodes which also contributes to this result. We also measured the percentage increase of the payload delivery with PIES in this case and we found it to be 68% over the case with AODV alone.

With higher ST:WT ratio, we can get better energy performance, i.e. more energy savings, which results in even longer network lifetime. To see this, we repeated the same experiment with ST/WT=0.5/0.5. The first node in this case started to die around 568.5 seconds with total operation stopping at around 593.7 seconds, which represents a roughly 69% lifetime extension. As for the percentage increase of payload delivery, we found it to be 79.5% over AODV alone in this case.

5.9 Comparison with the On-demand Power Management Algorithm

In this section, we compare the performance of the PIES algorithm with that of the on-demand power management algorithm [49]. We have selected this algorithm for our comparison since it both addresses the main source of energy waste, idle energy consumption, and its operation is based on a realistic network model. The on-demand power management algorithm selects a routing backbone that remains turned on, based on the route discovery strategy of the routing protocol. We use DSR as the routing protocol in this comparison experiment.

The DSR routing protocol belongs to the reactive routing category. Its functionality depends on including the full route to a destination in the header of each packet intended for this destination. Each node may keep several routes to a given destination in its route cache. The route selection is based on the shortest path. DSR uses promiscuous mode to enhance its path discovery and maintenance capabilities. This way, nodes can learn about routes to different destinations and include this route information in their routing cache. As far as PIES is concerned, it is important to see the effect of node sleep modes on their ability to acquire such additional route knowledge and therefore the effect on DSR’s performance with PIES enabled.

5.9.1 Qualitative Comparison

In order to pave the way for the experimental comparison between the two algorithms, we first perform a brief comparison between the functionality of both PIES and the on-demand power saving algorithm which we reviewed in Section 3.3. The following table compares the main features of both algorithms:

Table 5-3: Comparison between PIES and On-demand power saving algorithm

	PIES	**On-demand**
Use of broadcast messages	Optional	Beacon messages
Dependent on the nature of the routing protocol	No	Yes
Method of determining neighbor sleep state	Deterministic	Inference
Energy fairness by design	Yes	No
Modularity	Yes	No
Routing protocols it can operate with	Any	Any
Separation between traffic nodes and routers	No	No

We see from this comparison that PIES provides a method to determine the node sleep state that is far more robust than that of the on-demand power saving algorithm. Also unlike the on-demand power saving algorithm, PIES has a modular architecture that enables it to fit easily with existing routing algorithms. Moreover, this structure does not need to be changed with the change of the routing algorithm in use. PIES also does not require supporting broadcast messages to be sent separately, which means that it does not impose additional traffic burden on the network.

5.9.2 Performance Evaluation

In order to compare the operation of PIES to that of the on-demand routing protocol, we selected the ST/WT pair value for PIES to be equal to 0.3/0.5. This is to achieve energy savings that are comparable to that of the on-demand algorithm in static scenarios, so that we do not focus the discussion solely on the amount of achievable energy savings.

We measured both the energy performance and network operation performance in our comparison. As far as energy performance is concerned, and for the selected ST/WT value of PIES, we found that PIES results in consistent power savings of about 35%

regardless of the mobility conditions of network nodes. The savings that PIES provides are higher than those of the on-demand power management scheme until the network becomes almost static. At static conditions, the on-demand algorithm provides marginally higher savings (about 37%), see Figure 5-29. As we have seen before, PIES can provide even higher savings than the on-demand scheme, including in the static case, if we increase the ST:WT ratio.

Considering energy consumption balance and fairness, the on-demand power management algorithm does not seem to pay any attention to this aspect. It seems that the nodes that are on some routes are severely penalized and are kept on for most if not all time while the other, less strategically positioned nodes, are allowed to enjoy extensive energy savings. This strategy can be detrimental to network operation in such cases where some nodes that are critical to the operation of the whole network are over-utilized and hence get depleted much faster than others. In the case of PIES, it shows energy performance that is fairest considering DSR alone and DSR with on-demand power management, see Figure 5-30.

As far as packet delivery is concerned, both PIES and the on-demand algorithm seem to perform comparably to DSR alone especially at lower mobility, see Figure 5-31. Considering packet delivery latency, we found that the on-demand power management algorithm performs clearly better than PIES, see Figure 5-32. The reason is that it keeps the nodes that are on active routes always on with no attention given to conserving their idle energy. Therefore, it can provide better packet delivery latency performance at the price of wasting energy and in an unbalanced manner for some nodes.

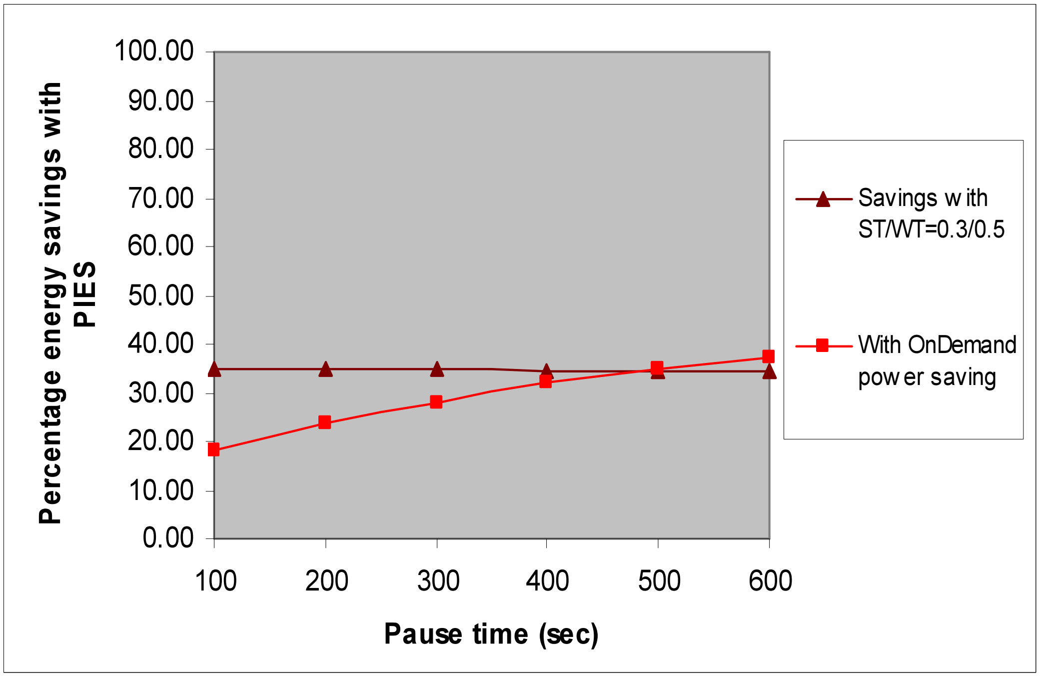


Figure 5-29: Energy Savings for PIES and On-demand power management over DSR alone

5.9.3 Effect on Network Lifetime

In order to complete the comparison, we also examined the performance of both PIES and the on-demand power saving algorithm under reduced initial energy conditions. For this purpose, we performed the above experiment with initial node energies of 300 Joules. We examined the effect on network lifetime. Figure 5-33 shows the results. From this figure, we see that PIES performs clearly better than the on-demand power saving algorithm even in the static network case.

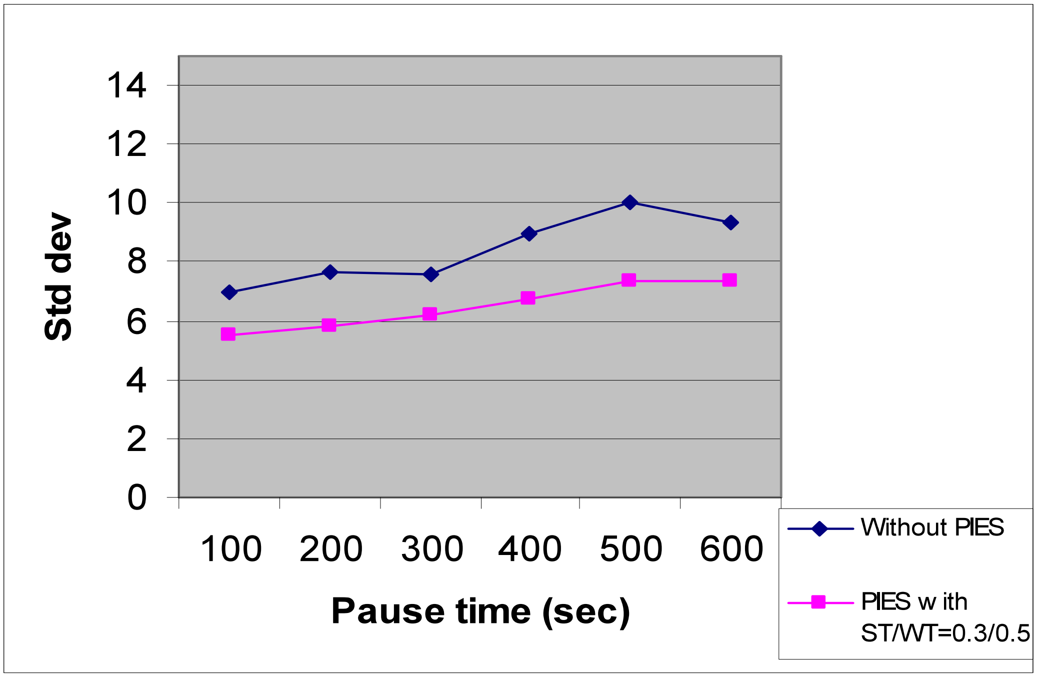
Since network lifetime extension alone does not give a sufficient indication about the network health in low energy conditions, we also examined the effect on network capacity. For this, we measured the number of delivered payload packets for both PIES and on-demand power saving relative to DSR alone. Figure 5-34 shows this comparison.

The results are rather interesting. It seems that, despite the longer lifetime that the on-demand algorithm gives over DSR alone, the corresponding increase in packet delivery is quite low especially at high mobility conditions. The increase in delivery is well below 10% at its highest point. In the case of PIES, the increase in packet delivery over DSR in these conditions is considerably higher and reaches above 60% in the static case. In high mobility conditions it is still high and is close to 50%.

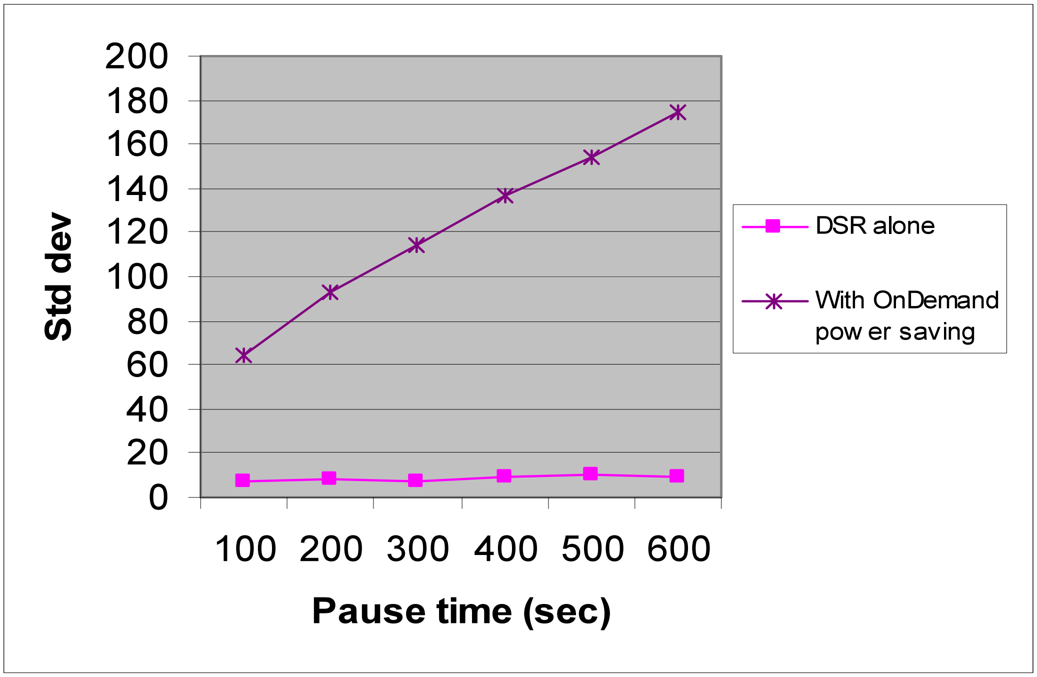
From this comparison, we see that overall, PIES performs clearly better than the on-demand power saving algorithm. It provides consistent power savings regardless of network mobility conditions. It does so in a fair manner to all network nodes which results in healthy network conditions for a considerably longer period of time over the on-demand power saving algorithm.

5.10 Summary

In this chapter, we presented some simulation results to demonstrate the functionality of the PIES algorithm and its effect on the energy and network operation performance. First, we showed that PIES performs equally well with the AODV and OLSR routing protocols which belong to the reactive and proactive categories of routing algorithms, respectively. We showed that PIES does not affect the core functionality of the routing algorithm. Simulations have shown that PIES introduces significant energy savings that reach about 50%. It does not achieve this at the expense of the energy balance between nodes and it actually enhances this balance. Simulations have also shown that with PIES enabled, packet delivery ratios are comparable to those achieved with the routing protocol alone (PIES disabled). Our experiments have shown that PIES has some effect on packet



(a)



(b)

Figure 5-30: Standard deviation for node energies: (a) in case of DSR alone, and with PIES, (b) DSR alone and with the On-demand power saving algorithm

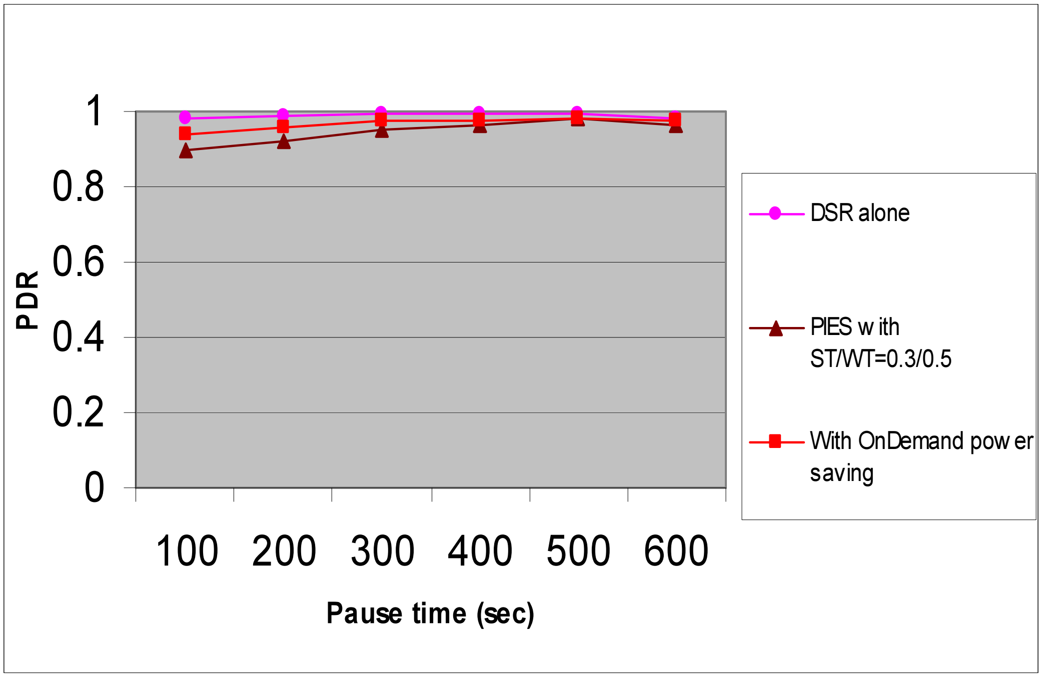


Figure 5-31: Packet delivery ratio performance with PIES and on-demand power management

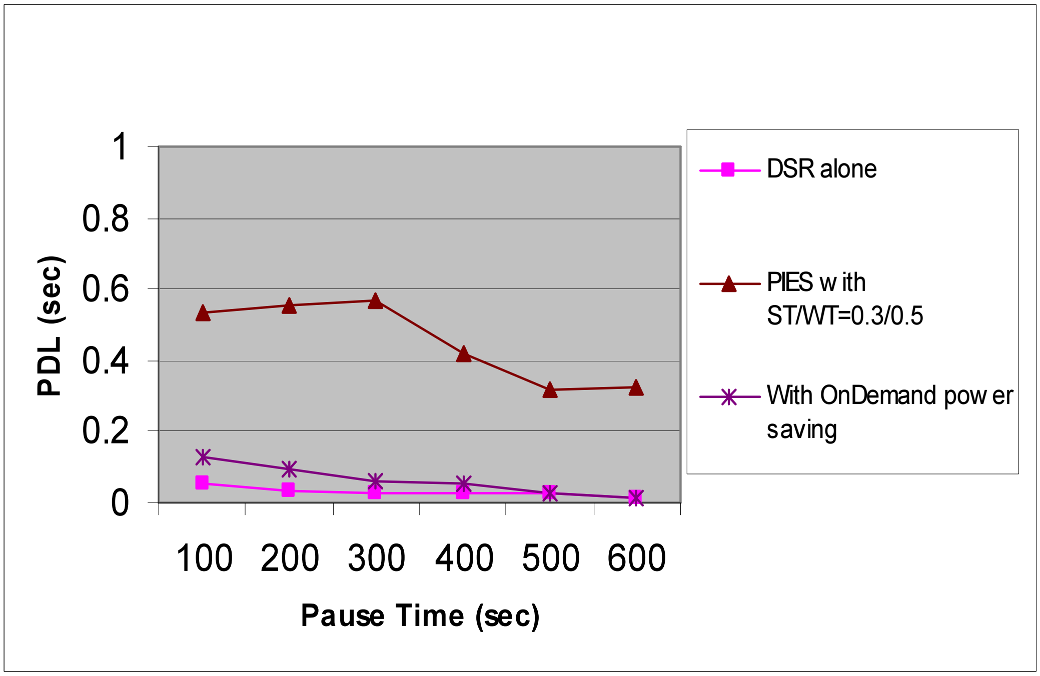


Figure 5-32: Packet delivery latency for DSR, PIES and on-demand power management

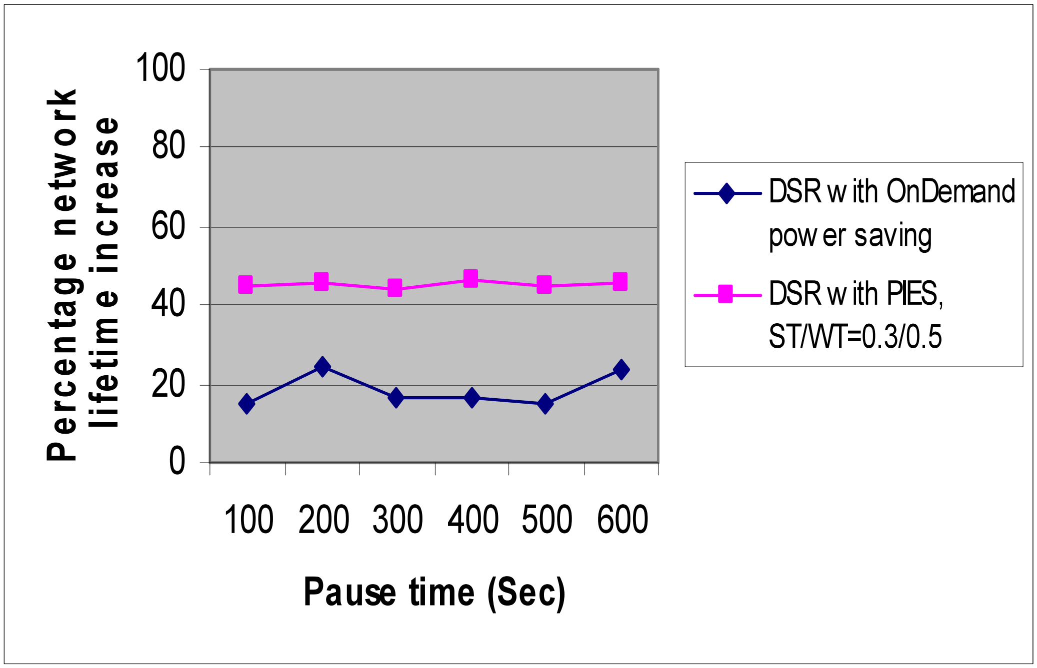


Figure 5-33: Percentage increase in network lifetime for PIES and on-demand power saving algorithms over DSR alone

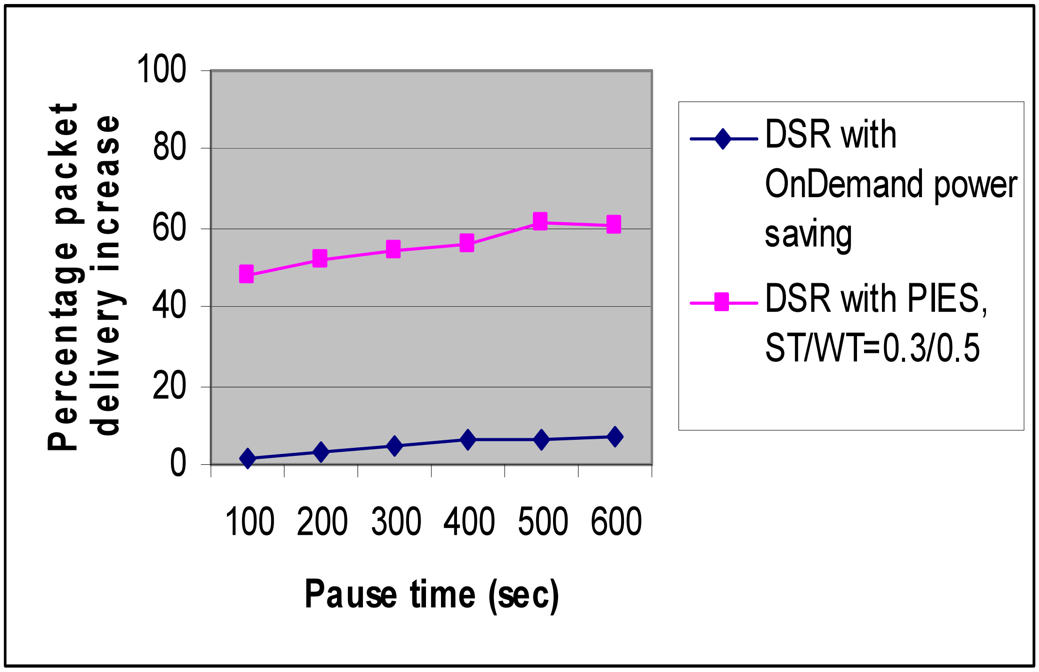


Figure 5-34: Percentage packet delivery increase for PIES and on-demand power saving over DSR alone in reduced energy conditions

latency. We also demonstrated the ability of PIES to scale well with increased network scale both from node population and traffic level perspectives. Finally, PIES performs well with bursty traffic which introduces new challenges to network operation.

We also tested the effect of changing PIES parameters on energy and network performance. This included experimenting with different ST/WT pair values that have the same ST:WT ratio. Our simulations showed that with these ST/WT values and the chosen load conditions, the energy performance is almost the same for all ST/WT values while the PDR and packet delivery latency performance is better with smaller ST values. These tests also involved experimenting with different values of the sleep separation factor. Our experiments showed that energy performance and network operation performance are generally insensitive to the value of this parameter.

We tested the effect that PIES has on extending the network lifetime. The routing algorithm alone failed to keep the network functioning beyond a certain point of time due to node failures while with PIES enabled, the network continued to function normally for a considerably longer period that reached almost 70% in our simulations. The percentage increase in delivered payload traffic reached almost 80%.

We also performed qualitative and experimental comparisons with the on-demand power saving algorithm with DSR as the routing protocol. Overall, PIES showed superior results to those of the on-demand routing protocol and was able to expand network life time far beyond the on-demand power saving case. It was also able to show a much better packet delivery performance under limited battery energy conditions.

When we consider these results, we find that PIES produces good network operation performance results with substantial energy savings and network lifetime extension. This is done while achieving energy fairness which is our primary target. PIES also integrates nicely with existing routing protocols and does not impose any unrealistic conditions on network operations (i.e. separating traffic nodes from routing nodes).

Chapter 6: Delay and Throughput Analysis of PIES

6.1 Introduction

Introducing an energy conserving strategy that depends on putting network nodes to sleep for periods of time has implications on important aspects of their operation. The most visible effect is on the packet delivery latency and on the throughput.

In the previous chapter, we ran extensive experiments that enabled us to measure the effect that PIES has on such network operation aspects as latency and packet delivery ratio under the conditions of these experiments.

In this chapter, we analyze the additional delay that PIES adds to the original packet delivery latency. We also perform a comparison between our analytical results and the results that we obtained through our simulation experiments. We also discuss analytically the effect of PIES on throughput in saturation conditions. In this throughput analysis, we consider the IEEE 802.11 MAC protocol. We also perform simulation experiments to verify our throughput analysis. Finally, we discuss the effect of the PIES ST and WT parameters on the overall performance.

6.2 PIES Delay Analysis

Since PIES introduces sleep periods, it is expected that the delay in sending packets between different nodes will generally be affected. Each node within the network operates on a cycle that is composed of a sleep time of duration ST seconds followed by a wakeup time of duration WT seconds. To analyze the additional end-to-end packet delivery latency that is introduced by PIES, let us first focus on any two communicating nodes X and Y within the network. We assume a no-congestion state of the network and that there is no contention on the communication channels for the nodes X and Y. We also assume the following [18]:

S is the event that both X and Y are awake,

$\overline{S}$ is the event that either of X or Y (or both) is asleep,

A is the event that X is awake,

$\overline{A}$ is the event that X is asleep,

B is the event that Y is awake, and

$\overline{B}$ is the event that Y is asleep

Note that the two events A and B are independent of each other. If D is the extra packet latency that is introduced by PIES between X and Y then, the expected value of D, E(D), is given by the following:

$$E(D) = E(D|S)\, P(S) + E(D|\overline{S})\, P(\overline{S}) = E(D|\overline{S})\, P(\overline{S}) \tag{6-1}$$

Since:

$$P(S) = P(A \cap B) = P(A|B)\, P(B) = P(A)\, P(B) = WT/(WT + ST) \times WT/(WT + ST)$$
$$= WT^2 / (WT + ST)^2$$

Therefore,

$$P(\overline{S}) = 1 - P(S) = 1 - WT^2 / (WT + ST)^2 \tag{6-2}$$

Substituting from (6-2) into (6-1):

$$E(D) = E(D|\overline{S}) \times [1 - WT^2 / (WT + ST)^2] \tag{6-3}$$

As mentioned above, the event $\overline{S}$ occurs in one of the following three cases:

- Both X and Y are asleep
- X is asleep
- Y is asleep

In order to evaluate the expected value of the delay in each of the above cases, we can use the cycle of one of the nodes, say X, as a reference without loss of generality, see Figure 6-1. We use a moving point of time, t_o, to point to the different possibilities of the event $\overline{S}$.

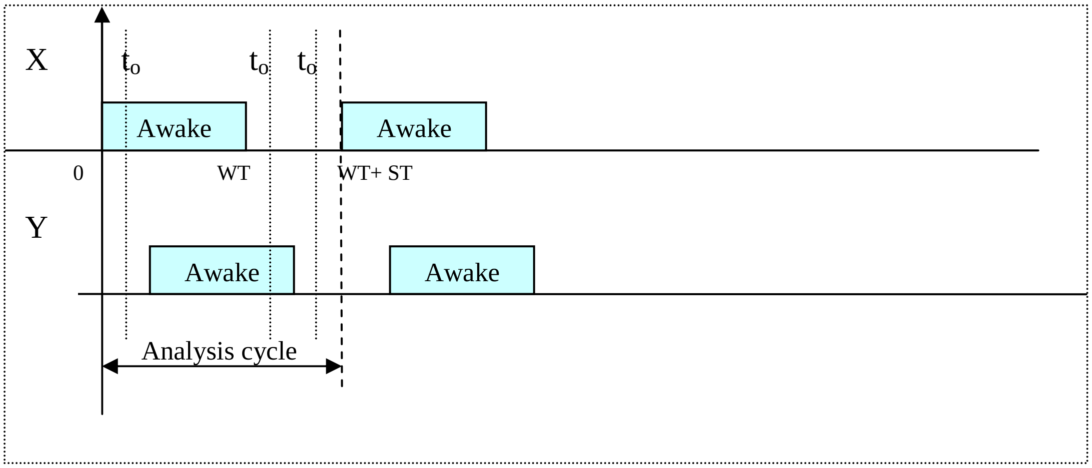


Figure 6-1: Sleep/Wakeup cycle of two communicating nodes, X and Y

Therefore, the expected value of the delay given $\overline{S}$ can be calculated as follows:

$$E(D|\overline{S}) = E(D_1)\,P(0 < t_o < WT) + E(D_2)\,P(WT < t_o < ST + WT) \tag{6-4}$$

where $E(D_1) = E(D|\overline{S},\ 0 < t_o < WT)$, and,

$E(D_2) = E(D|\overline{S}, WT < t_o < ST + WT)$

In the first period, $0 < t_o < WT$, the delay will happen only if Y is asleep. Since the sleep event is uniformly distributed U[0 , ST], therefore[30]:

$$E(D_1) = ST/2 \tag{6-5}$$

In the period ($WT < t_o < ST + WT$), delay will happen in two cases:

D_{21}: X asleep and Y asleep

D_{22}: X asleep and Y awake

Therefore,

$$\begin{aligned} E(D_2) &= E(D_{21})\,P(D_{21}) + E(D_{22})\,P(D_{22}) \\ &= E(D_{21})\,P(\overline{A} \cap \overline{B}) + E(D_{22})\,P(\overline{A} \cap B) \\ &= E(D_{21})\,P(\overline{A})\,P(\overline{B}) + E(D_{22})\,P(\overline{A})\,P(B) \\ &= ST/2 \times [ST/(ST+WT)]^2 + ST/2 \times ST \times WT/(ST + WT)^2 \\ &= ST^2/[2\times(ST + WT)] \end{aligned} \tag{6-6}$$

Substituting from (6-5) and (6-6) into (6-4):

$$E(D|\overline{S}) = ST/2 \times WT/(ST + WT) + ST^2/[2\times(ST + WT)] \times ST/(ST + WT)$$

$$= \{ST/[2(ST + WT)]\}\ \{ WT + ST^2/(ST + WT)\} \qquad (6\text{-}7)$$

Substituting from (6-7) into (6-3), the expected value of the delay between X and Y can be obtained as follows:

$$E(D) = \{ST/[2(ST+WT)^2]\}\ \{WT^2 + WT \times ST + ST^2\}\{(ST^2 + 2\ WT \times ST)/(ST + WT)^2\}$$

$$= \{ST^2/2(ST + WT)^4\}\{ST + 2WT\}\{WT^2 + ST \times WT + ST^2\} \qquad (6\text{-}8)$$

Equation (6-8) gives the expected value of the latency introduced by PIES for each packet exchanged between any two nodes within the network. Therefore, to calculate the additional end-to-end latency that is introduced by PIES for a packet exchanged between a source and destination with (n) intermediate nodes in between, we use the following equation:

$$E(D_{e\text{-}e}) = (n+1)\ \{ST^2/2(ST + WT)^4\}\{ST + 2WT\}\{WT^2 + ST \times WT + ST^2\} \qquad (6\text{-}9)$$

where $D_{e\text{-}e}$ is the additional delay introduced by PIES. This value does not include the original delay as obtained in the case of the routing algorithm alone. The above derivation also assumes that each two communicating nodes will remain in range of each other for the duration of this packet exchange.

Using equation (6-9), one can obtain the expected PIES delay between a source and a destination when using different values of sleep and wakeup times with different numbers of intermediate nodes, see Figure 6-2. We notice from this figure that as far as PIES sleep and wakeup times are concerned, the delay increases with the increase of the sleep time. For the same sleep time, the delay also increases with the increase of the ST:WT ratio. Therefore, for the same ST:WT ratio, the delay is lower with a lower value of ST. This is consistent with the experimental observations that we obtained previously. Since energy savings are the same for the same ST:WT ratio as we have seen in previous experiments, this gives us an indication that in general, smaller sleep times are preferable to achieve the same energy savings with lower latencies.

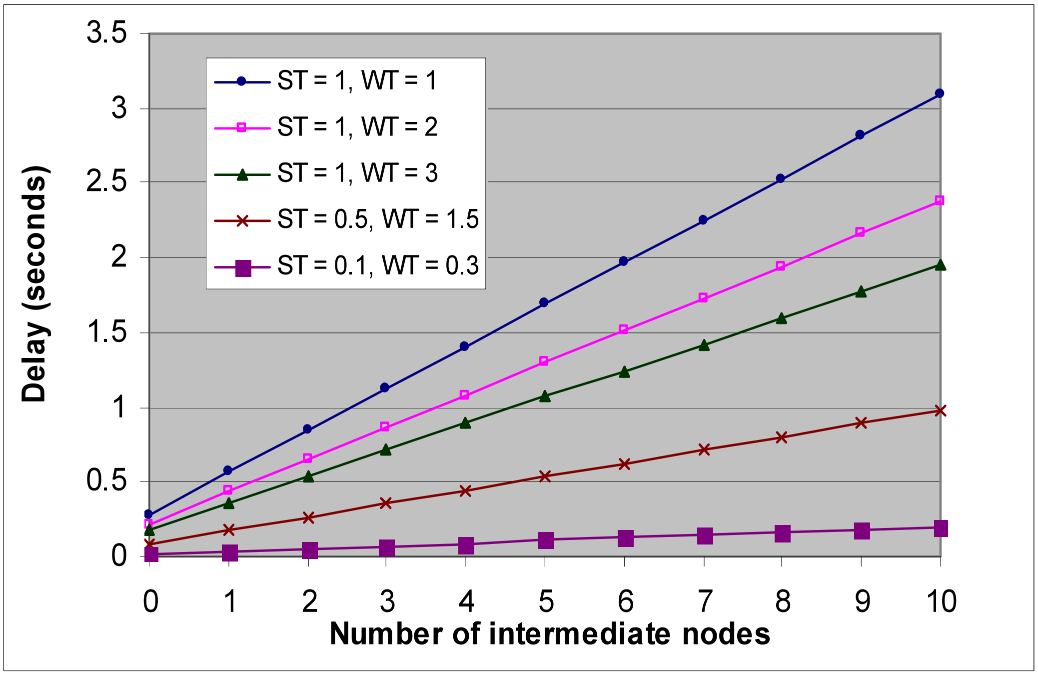


Figure 6-2: PIES expected delay for different ST/WT values

In order to compare the analytically obtained results against the ones obtained experimentally, we ran PIES with AODV in the static case with simulation conditions as in Section 5.4.1. Figure 6-3 shows the comparison between the results obtained in both cases for different numbers of intermediate nodes between the source and destination. We notice from this figure that the curves are following the same trend. If we consider the curve that was obtained through our analysis and the curve that was obtained through experiments, we find the experimental curve to be generally higher than the mathematical curve. This is attributed to the fact that the delay in the experimental case is the sum of original routing protocol end-to-end delay plus the delay introduced by PIES, while the analytical case shows only the PIES delay. For this reason, we also included in this figure the approximate trend for the delay that was introduced by PIES only in the experimental case. We calculated this trend from the experimental cases where AODV was active

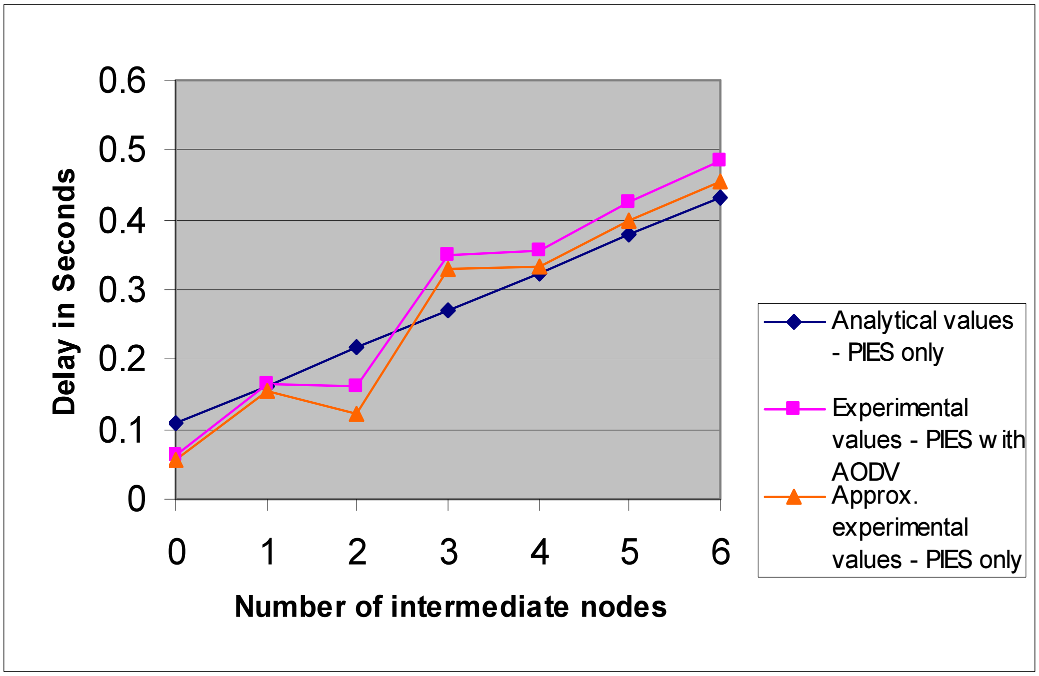


Figure 6-3: Analytical versus experimental delay with PIES for ST/WT=0.25/0.5

alone without PIES and then with PIES enabled. This trend shows to be generally close to the one that we obtained analytically.

6.3 PIES Saturation Throughput Analysis

In previous studies [4],[5],[51], the throughput of the IEEE 802.11 MAC protocol has been studied. In these studies, the case where *n* stations are contending for the communication channel is considered. These studies have considered the case where all nodes have traffic to send with each node being ready to send traffic immediately once it gets the chance to transmit on the communication channel. Therefore, these studies did not consider the case where energy saving algorithms are enabled with the strategy of putting nodes to sleep to reduce idle energy consumption. In order to accommodate this situation, this analysis needs to be modified in order to consider the case where one or more of the n contending nodes go to sleep. We notice that the analysis in all these studies follow the same procedure with results varying based on the state transition cases considered. In the following analysis, we follow the analysis of [51] in terms of its transition probabilities and we make our modifications based on the sleep model of the PIES algorithm. Let us assume the following:

- β is the probability that a node transmits during a slot time,
- τ is the probability that the backoff counter of a node equals zero at a certain slot time,
- W is the minimum size of the contention window,
- m is upper limit of the backoff delay stage,
- n is the number of contending nodes,
- p is the probability that a transmitted frame collides,

- p_b is the probability that the communication channel is busy, and,
- p_c is the probability that at least one connection is active in a given slot time.

Following the analysis of [51], the probability τ is defined as follows:

$$\tau = \frac{2(1-p_b)(1-2p)}{2(1-p_b)^2(1-2p)(1-p)+(p_b+p(1-p_b))(1-2p)(W+1)+pW(p_b+p(1-p_b))(1-(2p)^m)} \quad (6\text{-}10)$$

When PIES is enabled, a node transmits when all of the following three events take place:

A: The sending node itself is awake, this event has a probability WT/(WT+ST)

B: The receiving node is awake, this event also has a probability WT/(WT+ST)

C: The backoff window is equal to zero, this event has a probability τ which is calculated by equation (6-10).

Events A and B as well as B and C are clearly independent. However, we will use the approximation that events A and C are also independent for simplicity. This can be justified from the perspective that the sending node being awake does not directly imply that the backoff window will be zero and that to reach this stage is independent from when the node becomes awake. Therefore, the probability that a node transmits during a time slot now becomes:

$$\beta = P[\text{transmission during a slot time}] = \tau \times WT^2/(WT + ST)^2 < \tau \quad (6\text{-}11)$$

This has implications on the probability of collision. Since a transmitted packet collides when two or more nodes transmit during a time slot, the probability of collision, p, is given by:

$$p = 1 - (1-\beta)^{n-1} \tag{6-12}$$

A communications channel is detected busy when at least one node transmits during a slot time. Therefore, the probability p_b that a channel is detected busy is given by:

$$p_b = 1 - (1-\beta)^{n} \tag{6-13}$$

Substituting equations (6-11), (6-12) and (6-13) into (6-10), we get an equation in one unknown, τ. Solving this equation we get τ. Knowing τ, ST and WT, we can calculate p_b. These values will help us calculate the throughput as follows. A transmission is considered successful when only one node transmits, given that there is at least one transmission. Therefore the probability P_s that a transmission is successful is given by:

$$P_s = \frac{n\beta(1-\beta)^{n-1}}{1-(1-\beta)^n} \tag{6-14}$$

When PIES is enabled, a connection is active when both ends of the connection are awake. The probability, p_c, that at least one connection is active during a certain time slot is given by:

$$p_c = 1 - \{ 1 - (\frac{WT}{WT+ST})^2\}^n \tag{6-15}$$

The mean number $E[\psi]$ of consecutive idle slot times before a transmission takes place is therefore given by:

$$E[\psi] = \frac{1}{p_b p_c} - 1 \tag{6-16}$$

As in [4],[5],[51], it is sufficient to calculate throughput between two consecutive transmissions. Therefore the normalized throughput, S, can be calculated as follows:

$$S = \frac{E[T]}{E[\Delta t]}$$

where T is the time used for successful transmission in an interval and Δt is the length between two consecutive transmissions. Therefore,

$$S = \frac{P_s E[P]}{E[\psi] + P_s T_s + (1 - P_s) T_c} \tag{6-17}$$

where E[P] is the average payload length, T_s is the average time that the channel is captured by a successful transmission and T_c is the average time where the channel is captured by a collision. Assuming that payload packets have the same length, E[P] = P, where P is the number of bits per packet. T_s and T_c are defined according to [4] as follows for RTS/CTS CSMA/CA:

$$T_s = RTS + \delta + SIFS + CTS + \delta + SIFS + H + P + \delta + SIFS + ACK + \delta + DIFS \tag{6-18}$$

$$T_c = RTS + \delta + DIFS \tag{6-19}$$

where $H = MAC_{hdr} + PHY_{hdr}$ is the frame header, δ is the propagation delay, SIFS is the short interframe space and DIFS is the distributed interframe space. RTS, CTS and ACK in the last two equations are the times taken for transmitting RTS, CTS and ACK packets, respectively. Note that since E[ψ] is measured in slot times, both T_s and T_c in equation (6-17) have to be calculated in slot times as well.

In order to perform our comparison between when PIES is active and when it is not, we simply replace β by τ in equations (6-12), (6-13) and (6-14) for the case when PIES is not enabled. We use the parameters as shown in Table 6-1. For the case where PIES is enabled, we use several values of the ST/WT pair in order to gain an understanding of the effect of the choice of these parameters and their ratio on the throughput performance. The values that we use for our comparison are: 0.25/0.5, 0.75/1.0 and 0.5/0.5.

With these parameters, and for the ST/WT values as specified above (in the case where PIES is enabled), we solve equations (6-10), (6-11), (6-12) and (6-13) to get β (τ in the case where PIES is disabled) and p_b. Consequently, we can get T_c and T_s and hence the throughput S for different numbers of nodes. We solve these equations for the initial backoff window value, W=16 and max retry stages =4 (i.e. m=3). Figure 6-4 shows the trend in the case where PIES is disabled and when it is enabled for all three ST/WT values that we selected. We notice from this figure that the saturation throughput tends to be almost the same for the case when PIES is disabled and in all cases when PIES is enabled.

Table 6-1: Throughput analysis parameters

Payload packet size (Bytes)	512
Mac header size (Bytes)	24
Physical header size (Bytes)	24
ACK packet size (Bytes)	38
RTS packet size (Bytes)	44
CTS packet size (Bytes)	38
SIFS (μ sec)	10
DIFS (μ sec)	50
Δ (μ sec)	1
Slot time (μ sec)	20
Channel bit rate (Mbps)	2
H (Bytes)	48

We also repeat the same procedure for W=32 and m=5 to see the effect. Figure 6-5 shows the results. This figure shows that in this case, the throughput behavior continues to follow the same trend in general as with the case of W=16 and m=3.

By examining the elements of the throughput equation, (6-17), we find that the effect that PIES has on the throughput value is represented by β, which is calculated by equation (6-11), as well as the probability p_c as calculated by equation (6-16). From this we can see that fixing the ST:WT ratio yields the same throughput value. Therefore, the saturation throughput is not affected by the change of the ST and WT parameters as long as the ST:WT ratio remains the same.

In order to measure the throughput trend experimentally to verify our findings, we ran simulation experiments with conditions as per Section 5.2 with AODV as the routing protocol. We modified these conditions though in such a way that enables us to simulate saturation conditions as follows. We used a simulation area of dimensions of 175 × 175 square meters. This is to ensure that all nodes are within range from each other as the nominal wireless range of each node is 250 meters. We run the experiments where we have different numbers of nodes sending traffic simultaneously. We use the cases of 10, 20, 30, 40 and 50 traffic sending nodes. Each node sends traffic at the rate of 500 Kbps to ensure that nodes always have packets to transmit. For each case, we ran the simulations with five randomly generated mobility scenarios. We use the same MAC parameters as mentioned above, with W=32 and m=5. When PIES is enabled, we use three ST/WT values: 0.25/0.5, 0.75/1.0 and 0.5/0.5. We ran these simulations with static network conditions. With these conditions, we measured the throughput trend in all cases. Figure 6-6 shows the resulting trend. We notice that the experimental results show throughput values that are lower than those shown by the corresponding theoretical results as shown in Figure 6-5. This can be attributed to the overhead of the routing protocol, which was not considered while devising the analytical model. This is also evident by the drop of the throughput as the number of nodes increase, which is due to the fact that the routing protocol overhead (e.g. route discovery messages) increases with the increase of network congestion. This occurs when nodes misinterpret delays that result from this severe congestion as route breakages and start flooding the network with route requests in an attempt to fix the routes that are thought to be broken. In case of PIES, the saturation throughput is even lower due to the fact that when nodes go to sleep in these congestion conditions, the routing overhead becomes even heavier due to routes that are thought to have been broken more frequently leading to a further decrease in the saturation throughput than in the case with PIES disabled.

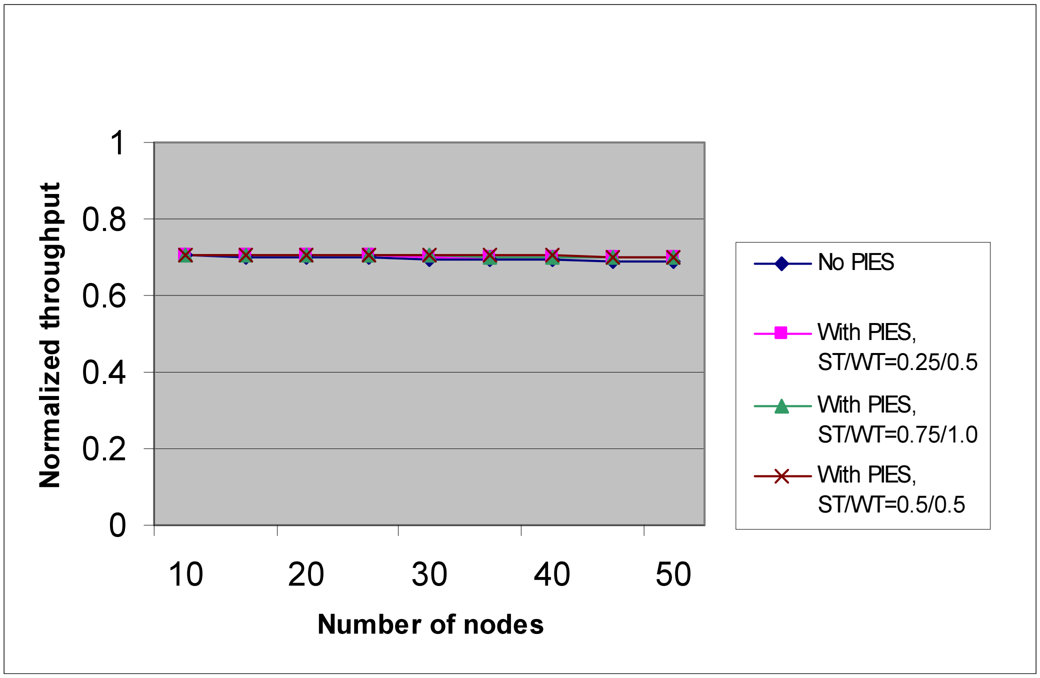


Figure 6-4: Throughput trend without PIES and with PIES enabled, W=16 and m=3

6.4 Selection of PIES ST and WT Parameters

The choice of the mandatory sleep time, ST, and mandatory wakeup time, WT, parameters has a profound effect on the performance of the system when PIES is enabled. The evaluation experiments in the last chapter as well as the analysis that we performed in this chapter have given us a good insight into the effect of the choice of these parameters on performance. It is clear that fixing the ST:WT ratio results in the same energy performance. On the other hand, as this ratio increases, we get a better energy performance with comparable packet delivery ratio. This, however, results in higher packet delivery latency which is the most visible side effect of increasing the ratio.

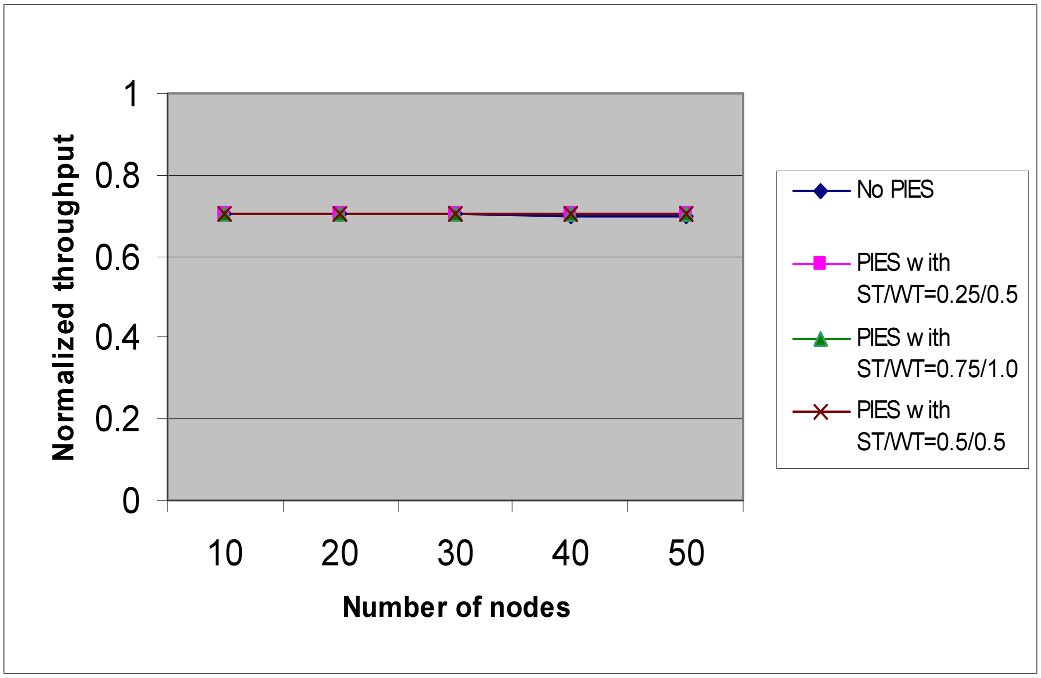


Figure 6-5: Throughput trend without PIES and with PIES enabled, W=32 and m=5

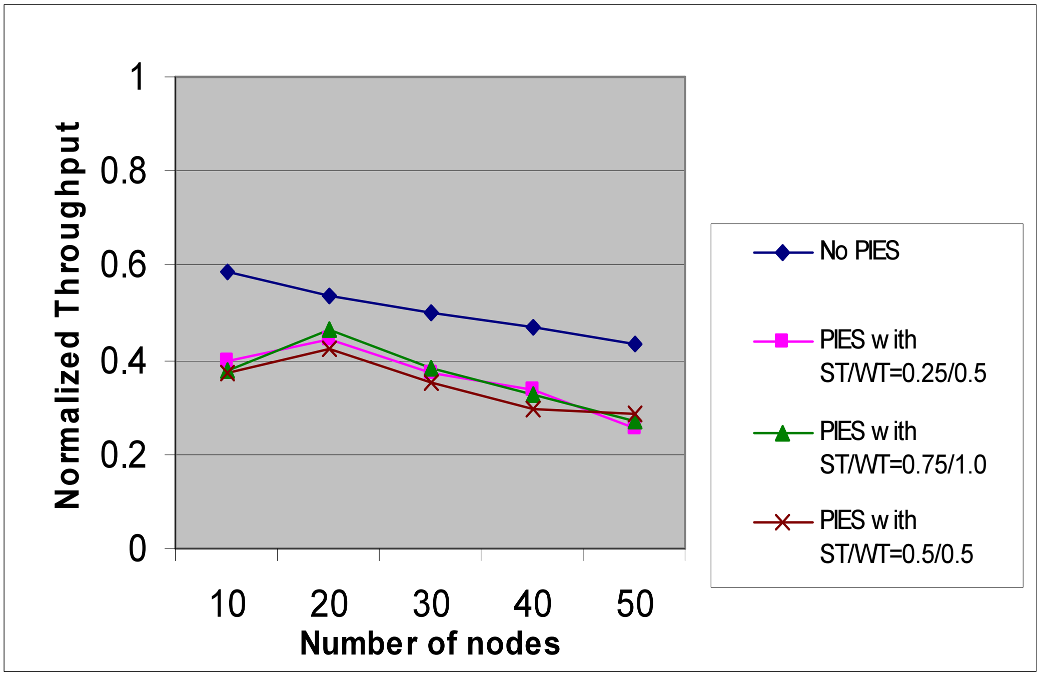


Figure 6-6: Experimental throughput trend without PIES and with PIES enabled

Packet delivery latency also decreases with the decrease of the value of the ST parameter for the same ST:WT ratio. Therefore, the choice of these parameters highly depends on the needs of the application at hand. For example, if it is important to enhance energy performance without worrying about latency, then increasing the ST:WT ratio is the way to go. If the aim is to keep latency below specific limits while having moderate energy performance at the same time, then reducing the ST:WT ratio is the right choice. However, we can always strike some balance if we want a certain level of energy savings while still keeping latency at reasonable levels. This can be achieved by experimenting to determine the ST:WT ratio that can achieve this level of energy savings and then fix this ratio and start working on decreasing the ST value until we reach an acceptable latency value. Generally speaking, this strategy can be implemented but in some cases the two goals may prove challenging to achieve at the same time.

As far as the lower limit of these parameters is concerned, it is obvious that we should keep the nodes awake for enough time to allow the transmission of at least one payload packet. This means that the length of the schedule overlap between two communicating nodes should be long enough for this to occur.

6.5 Summary

In this chapter, we analyzed the effect of the PIES sleep time, ST, and wakeup time, WT, on two aspects of the performance, namely the delay and saturation throughput. We have derived an equation that can be used to calculate the expected value of the additional delay that PIES introduces. We showed the effect of the choice of the relative values of the ST and WT parameters on the packet delivery delay. We also compared the results that we obtained analytically with those obtained through our simulation experiments and showed that the results in both cases are close to each other.

For the saturation throughput analysis, we used the analytical results of previous studies of the IEEE 802.11 MAC saturation throughput and adapted it to incorporate the case where PIES is enabled in the network. Analytically, the saturation throughput was shown to have a constant value regardless of the number of contending nodes. This value was not affected by enabling PIES. Through experiments, however, we showed that the saturation throughput in general is lower than in the analytical case due to the additional routing protocol overhead. It also gets lower as the number of the contending nodes increases.

Finally, we discussed the effect of the choice of the ST and WT parameters on the overall performance and offered some suggestions for the strategy of choosing these parameters.

Chapter 7: Summary, Conclusions and Future Work

7.1 Summary and Conclusions

Recently, mobile ad hoc networks (MANET) and their applications have become quite popular with the proliferation of light-weight mobile devices that made it possible to communicate and perform many types of tasks while on the move. Many protocols have been developed to handle routing in ad hoc networks. Each of these protocols has been developed based on different design strategies with the purpose of obtaining the best possible performance and robust data delivery in an environment with potentially constantly changing topology. Many of these algorithms, however, have not considered one important aspect of the operation of this type of networks which is the generally limited amount of energy that is available to its nodes. This can be considered the most critical factor in the operation of these networks. In this thesis, we presented the main characteristics of the mobile ad hoc networks as well as the factors that affect their operation. Then, we described the energy efficiency issues that are encountered with this type of networks and supported this discussion with a case study. This case study showed that there exists a large amount of energy (more than 50% of the overall energy that is consumed in communication) that is wasted while the wireless interfaces of the network nodes are in idle mode. We then described some of the schemes that were devised to address energy efficiency issues in MANETs. We classified such schemes as routing and non-routing energy-efficient schemes. According to this classification, the routing schemes are either energy-efficient routing algorithms or algorithms that directly influence the routing functionality of the routing protocol. The non-routing schemes, on the other hand, are those that do not directly affect the routing functionality of the routing protocol in use. We found from this survey that most of the routing-related energy-efficient schemes focus mainly on energy balance between routes and do not take idle energy consumption into consideration. The non-routing energy-efficient schemes had various strategies for addressing the idle energy consumption. Most of these schemes, however, did not pay attention to energy fairness, and some of them have been designed based on rather unrealistic assumptions about network operation. We then presented our solution to address the energy efficiency problem. We call our new technique PIES, which stands for *Protocol Independent Energy Saving*. PIES was designed with energy fairness central to its operation. It is not a routing algorithm. It integrates with existing routing algorithms to complement their functionality and enhances the overall energy efficiency of the node. PIES can be configured in such a way that it imposes no additional energy or traffic cost on the network. In addition, it does not affect the core functionality of the routing algorithm. In addition, PIES is modular in design and can therefore be integrated easily with routing protocols of different strategies. PIES does not depend on one node or a set of nodes for its functionality. Rather, its functionality is fully distributed, which ensures a robust functionality of the algorithm. PIES operation is based on a realistic network model and does not make any special assumptions about the network configurations or the functionality of its nodes.

In order to evaluate the functionality of our new scheme, we ran simulation experiments to demonstrate its performance from different perspectives. First, we showed its independence of the routing strategy by demonstrating its ability to function equally well with routing algorithms that belong to the proactive and the reactive categories of routing. We then showed its ability to scale well by showing that its performance is not affected by the increase in network population as well as network traffic. We showed that PIES performs well with bursty traffic conditions which introduce many challenges to network operation. We also experimented with the effect of changing PIES parameters on the network performance. We demonstrated PIES' ability to extend the network lifetime and showed that it can extend it by about 70% in our experiments. Finally, we performed qualitative and experimental comparisons with the on-demand power saving algorithm. We showed that, overall, PIES outperforms the on-demand power saving algorithm from a combined energy performance and network operation performance points of view.
We then performed an analysis of the additional delay that PIES introduces to packet delivery. Through this analysis, we established a trend that enabled us to understand the effect that the choice of combination of the PIES sleep time and wakeup time has on the resulting delay. We also analytically studied the effect that PIES has on network saturation throughput. We compared all our analytical results with the results that we obtained experimentally and we found that there is some difference due to the routing protocol overhead in the experimental case. Based on these analytical studies, we offered a discussion around the choice of the PIES parameters depending on the type and nature of the application at hand.

Based on our experiments and analysis, we conclude that PIES performs well from energy and network operation perspectives, independently of the type or nature of the routing protocol. It results in substantial energy savings and network lifetime extension. Its introduction to the network does not have a major impact on network operations. PIES also scales well with increased network traffic and population. Its performance is not affected by the type of traffic. It also has superior energy, network lifetime and data delivery performance compared to energy conservation techniques of the same category.

7.2 Suggestions for Future Work

For future investigation, the effect of introducing energy-efficient techniques that depend on putting nodes to sleep for periods of time on applications such as QoS and multicast needs to be investigated. In connection to PIES, methods to adapt the regular sleep/wakeup cycles to the requirements of the different applications need to be investigated such that they respond to the needs of such applications and at the same time will not compromise significantly the gains achieved by having predictable sleep cycles.

Also, the possibility of using duty cycles with different sleep/wakeup patterns per cycle instead of the current regular PIES operation cycles needs to be investigated. The utilization of duty cycles of this type, while adds complexity to the algorithm, may have the promise of achieving even higher levels of energy savings.

Another area to investigate is the derivation of methods to control latency that results from using sleep times as a means to conserve energy. For example, this can be done through investigating the possibility of influencing the routing protocol to route through nodes that have similar sleep schedules.

A modified saturation throughput model that takes into consideration the overhead of the routing protocol needs to be devised. This model needs to incorporate the needs of the routing protocol as well as any other component that adds overhead to the network operation.

The lower limits of the sleep and wakeup times for PIES need to be further analyzed. The effect of these lower limits on both the energy performance and network operation needs to be studied and determined for the different categories of the routing protocols.

Finally, since there are several algorithms that have been devised to affect the routing strategy only from an energy-efficiency point of view without a strategy to address idle energy consumption, an area worth investigating is the possibility of integrating PIES with such algorithms and measuring the resulting combined effect.

References

[1] J. Banks, J. S. Carson and B. L. Nelson, “Discrete-Event System Simulation”, Prentice-Hall, 1996, ISBN 0-13-217449-9.

[2] P. Bergamo, A. Giovanardi, A. Travasoni, D. Maniezzo, G. Mazzini and M. Zorzi, “Distributed Power Control for Energy Efficient Routing in Ad Hoc Networks”, Wireless Networks, Volume 10, No. 1, January 2004, pages 29-42.

[3] P. Bergamo, D. Maniezzo, A. Giovanardi, G. Mazzini and M. Zorzi, “Distributed Power control for Power-aware Energy-efficient Routing in Ad hoc Networks”, in Proceedings of European Wireless 2002, Florence, Italy, February 2002, pages 237-243.

[4] G. Bianchi, “IEEE 802.11-Saturation Throughput analysis”, IEEE Communications Letters 2(12), 1998, pages 318-320.

[5] G. Bianchi, “Performance Analysis of the IEEE 802.11 Distributed Coordination Function”, IEEE Journal on Selected Areas in Communications, Vol. 18, No. 3, March 2000, pages 535-547.

[6] J. Broch, D. A. Maltz, David B. Johnson, Yih-Chun Hu and Jorjeta Jetcheva, “Performance Comparison of Multi-hop Wireless Ad Hoc Network Routing Protocols”, Proceedings of the ACM Mobile Computing and Networking Conference, Dallas, Texas, October 1998, pages 85-97.

[7] T. X. Brown, H. N. Babow and Q. Zhang, “Maximum Flow-Life Curve for a Wireless Ad Hoc Network”, Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing, Long Beach, CA, USA, 2001, pages 128-136.

[8] J.-H Chang and L. Tassiulas, “Energy Conserving Routing in Wireless Ad-hoc Networks”, in Proceedings of INFOCOM, 2000, Tel Aviv, Israel, pages 22-31.

[9] S. Corson and J. Macker, “Mobile Ad Hoc Networking (MANET): Routing Performance Issues and Evaluation Considerations”, IETF RFC 2501, January 1999.

[10] B. Chen, K. Jamieson, H. Balakrishnan, and Robert Morris, “Span: An Energy-Efficient Coordination Algorithm for Topology Maintenance in Ad Hoc Wireless Networks”, in Proceedings of the ACM Mobile Computing and Networking Conference, Rome, Italy, 2001, pages 85-96.

[11] T. H. Clausen, G. Hansen, L. Christensen and G. Behrmann, “The Optimized Link State Routing Protocol, Evaluation through Experiments and Simulation”, Proceedings of the 4th IEEE International Symposium on Wireless Personal Mobile Communications, September 2001.

[12] The CMU Monarch Project’s Wireless and Mobility Enhancements to NS. http://www.monarch.cs.cmu.edu

[13] K. Fall and K. Varadhan, editors, “NS notes and documentation”, the VINT project, UC Berkeley, LBL, USC/ISI and Xerox PARC, July 29, 2000. http://www.isi.edu/nsnam/ns/ns-documentation.html

[14] L. M. Feeney, “An Energy Consumption Model for Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks”, Mobile Networks and Applications, Volume 6, Issue 3, June 2001, pages 239 - 249.

[15] L. M. Feeney and M. Nilsson, “Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment”, Proceedings of IEEE INFOCOM 2001, Volume 3, Anchorage AK, April, 2001, pages 1548-1557.

[16] Y. Gadallah and T. Kunz, "Energy consumption in ad-hoc routing protocols: Comparing DSR, AODV and TORA", Proceedings of the 1st International Conference on Ad-Hoc Networks and Wireless, Toronto, Canada, September 2002, pages 161-176.

[17] Y. Gadallah and T. Kunz, "PIES: Protocol independent energy saving algorithm", Proceedings of the 2004 Workshop on Mobile and Wireless Networking, Montreal, Canada, August 2004, pages 4-12, IEEE Computer Society Press 2004, ISBN 0-7695-2198-3.

[18] Y. Gadallah and T. Kunz, “A Protocol Independent Energy Saving Technique for Mobile Ad Hoc Networks”, accepted for the special issue on Mobile and Wireless Networking, International Journal of High Performance Computing and Networking (IJHPCN). Expected to be published early 2005.

[19] H. Hassanein and A. Safwat, “Virtual base stations for wireless mobile ad hoc communications: an infrastructure for the infrastructure-less”, The International Journal of Communication Systems, Vol. 14, 2002, pages 763-782.

[20] P. J. M. Havinga and G. J. M. Smit, “Energy-efficient wireless networking for multimedia applications”, Wireless Communications and Mobile Computing, Wiley, 2001, 1:165-184.

[21] T. Clausen and P. Jacquet, “Optimized Link State Routing protocol (OLSR)”, IETF RFC 3626, October 2003.

[22] Amit Jardosh, Elizabeth M. Belding-Royer, Kevin C. Almeroth, and Subhash Suri, "Towards Realistic Mobility Models for Mobile Ad hoc Networks", Proceedings of the ACM Mobile Computing and Networking Conference, San Diego, CA, September 2003, pages 217-229.

[23] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, “Scenario-based performance analysis of routing protocols for mobile ad hoc networks”, Proceedings of the ACM Mobile Computing and Networking Conference, Seattle, Washington, August 1999, pages 195-206.

[24] David B. Johnson, David A. Maltz, and Josh Broch, “DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks”, in *Ad Hoc Networking*, edited by Charles Perkins, Addison-Wesley 2001, ISBN 0-201-30976-9, pages 139-172.

[25] C. E. Jones, K. M. Sivalingam, P. Agrawal and J. C. Chen, “A Survey of Energy Efficient Network Protocols for Wireless Networks”, Wireless Networks, Kluwer Academic Publishers, 2001, pages 343-358.

[26] P. Karn, “MACA – a New Channel Access Method for Packet Radio”, in ARRL/CRRL Amateur Radio 9th Computer Networking Conference, London, Ont., Canada, 1990, pages 134-140.

[27] D. Kim, J. J. Garcia-Luna-Aceves, K. Obraczka, J.-C Cano and P. Manzoni, “Routing Mechanisms for Mobile Ad Hoc Networks Based on the Energy Drain Rate”, IEEE Transactions On Mobile Computing, Vol. 2, No. 2, April-June 2003, pages 161-173.

[28] R. Kravets and P. Krishnan, “Power Management Techniques for Mobile Communications”, Proceedings of the ACM Mobile Computing and Networking Conference, Dallas, Texas, October 1998, pages 157-168.

[29] A. Laouiti, P. Muhlethaler, A. Najid, E. Plakoo, “Simulation Results of the OLSR Routing Protocol for Wireless Network”, Proceedings of the 1st Mediterranean Ad-Hoc Networks Workshop (Med-Hoc-Net), Sardegna, Italy, 2002.

[30] A. Leon-Garcia, “Probability and Random Processes for Electrical Engineering”, Addison-Wesley, 1994, ISBN 0-201-50037-X.

[31] L. Leppänen, J. Prokkola and T. Bräysy, “Performance of Ad Hoc Network under Pareto Distributed Traffic Model”, Finnish Wireless Communication Workshop (FWCW 2003), Oulu, Finland, October 2003, pages 21-24.

[32] C. E. Perkins, “Ad Hoc Networking”, Addison-Wesley, 2001, ISBN 0-201-30976-9.

[33] C. E. Perkins and P. Bhagwat, “Highly Dynamic Destination-Sequenced Distance-Victor Routing (DSDV) for Mobile Computers”, Computer Communications Review, October 1994, pages 234-244.

[34] C. E. Perkins and E. M. Royer, “Ad-hoc On-Demand Distance Vector Routing”, Proceedings of 2nd IEEE workshop on Mobile Computing Systems and Applications, February 1999, pages 90-100.

[35] C. E. Perkins, E. M. Royer, S. R. DAS and M. K. Marina, “ Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks”, IEEE Personal Communications, February 2001, pages 16–28.

[36] J. Prokkola, L. Leppänen and T. Bräysy, “On the Effect of Traffic Models to Ad Hoc Network Performance”, Proceedings of the Military Communications Conference (MILCOM), Boston, USA, October 2003, pages 7-13.

[37] C.S. Raghvendra and Suresh Singh, “PAMAS – Power Aware Multi-Access protocol with Signaling for Ad Hoc Networks”, ACM Computer Communications Review, July 1998, Volume 28, Issue 3, pages 5-26.

[38] E. M. Royer and C-K Toh, “A review of current routing protocols for Ad-Hoc Mobile Wireless Networks”, IEEE Personal Communications, Volume 6, Issue 2, April 1999, pages 46-55.

[39] A. Safwat, H. Hassanein, and H. Mouftah, “Energy-Aware Routing in MANETs: Analysis and Enhancements,” Proceedings of The Fifth ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2002) in conjunction with the ACM Mobile Computing and Networking Conference, Atlanta, Georgia, USA, September 2002, pages 46-53.

[40] A. Safwat, H. Hassanein and H. Moftah, “Power-Aware Fair Infrastructure Formation for Wireless Mobile Ad hoc Communications”, Proceedings of IEEE GLOBECOM, San Antonio, Texas, USA, November 2001, pages 2832-2836.

[41] J. Schiller, “Mobile Communications”, Addison-Wesley, ISBN 0-201-39836-2, 2000, pages 182-186.

[42] S. Singh, M. Woo, and C. S. Raghavendra, “Power-aware Routing in Mobile Ad Hoc Networks”, Proceedings of the ACM Mobile Computing and Networking Conference, Dallas, Texas, October 1998, pages 181-190.

[43] M. Stemm and R. H. Katz, “Measuring and reducing energy consumption of network interfaces in hand held devices”, IEICE Transactions on Communications, E80-B(8):1125-1131, August 1997.

[44] I. Stojmenovic and X. Lin, “Power-aware localized routing in wireless networks”, IEEE Transactions on Parallel and Distributed Systems, November 2001, 12(11):1122-1133.

[45] A. S. Tanenbaum, “Distributed operating systems”, Englewood Cliffs, N.J., Prentice Hall, ISBN 0132199084, 1995.

[46] C.-K. Toh, “Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks”, IEEE Communications Magazine, June 2001, pages 2-11.

[47] Ya Xu, J. Heidemann and D. Estrin, “Geography-informed Energy Conversion for Ad Hoc Routing”, in Proceedings of the ACM Mobile Computing and Networking Conference, Rome, Italy, July 2001, pages 70-84.

[48] R. Zheng, J. C. Hou and Lui Sha, “Asynchronous Wakeup for Ad Hoc Networks”, in Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing, Annapolis, Maryland, USA, June 2003, pages 35-45.

[49] R. Zheng and R. Kravets, “ On-demand Power Management for Ad Hoc Networks”, in Proceedings of INFOCOM, San Francisco, CA, April 2003, pages 1-11.

[50] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999. IEEE 802.11 Standard, LAN MAN Standards Committee of the IEEE Computer Society.

[51] E. Ziouva and T. Antonnakopoulos, “CSMA/CA Performance under High Traffic Conditions” Throughput and Delay analysis”, Computer Communications, Volume 25 No. 3, 2002, pages 313-321.