Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

ENERGY-EFFICIENT MULTI-CRITERIA PACKET FORWARDING IN MULTI-HOP WIRELESS NETWORKS

by
Komlan Egoh

Reliable multi-hop packet forwarding is an important requirement for the implementation of realistic large-scale wireless ad-hoc networks. However, packet forwarding methods based on a single criterion, such as the traditional greedy geographic forwarding, are not sufficient in most realistic wireless settings because *perfect-reception-within-range* cannot be assumed. Furthermore, methods where the selection of intermediate relaying nodes is performed at the transmitter-side do not adapt well to rapidly changing network environments. Although a few link-aware geographic forwarding schemes have been reported in the literature, the tradeoffs between multiple decision criteria and their impact on network metrics such as throughput, delay and energy consumption have not been studied.

This dissertation presents a series of strategies aimed at addressing the challenges faced by the choice of relay nodes in error-prone dynamic wireless network environments. First, a single-criterion *receiver-side relay election* (RSRE) is introduced as a distributed alternative to the traditional transmitter-side relay selection. Contrary to the transmitter-side selection, at each hop, an optimal node is elected among receivers to relay packets toward the destination. Next, a multi-criteria RSRE, which factors multiple decision criteria in the election process at lower overhead cost, is proposed. A general cost metric in the form of a multi-parameter mapping function aggregates decision criteria into a single metric used to rank potential relay candidates. A two-criteria RSRE case study shows that a

proper combination of greedy forwarding and link quality leads to higher energy efficiency and substantial improvement in the end-to-end delay. Last, mesh multi-path forwarding methods are examined. A generalized mesh construction algorithm in introduced to show impact of a mesh structure on network performance.

ENERGY-EFFICIENT MULTI-CRITERIA PACKET FORWARDING IN MULTI-HOP WIRELESS NETWORKS

by
Komlan Egoh

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Engineering

Department of Electrical and Computer Engineering

May 2012

APPROVAL PAGE

ENERGY-EFFICIENT MULTI-CRITERIA PACKET FORWARDING IN MULTI-HOP WIRELESS NETWORKS

Komlan Egoh

Dr. Roberto Rojas-Cessa, Dissertation Advisor — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Nirwan Ansari, Dissertation Co-Advisor — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Stewart D. Personick, Committee Member — Date
Senior University Lecturer, Department of ECE, NJIT

Dr. Osvaldo Simeone, Committee Member — Date
Assistant Professor of Electrical and Computer Engineering, NJIT

Dr. Guiling Wang, Committee Member — Date
Assistant Professor of Computer Science, NJIT

BIOGRAPHICAL SKETCH

Author: Komlan Egoh

Degree: Doctor of Philosophy

Date: May 2012

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Engineering,
  New Jersey Institute of Technology, Newark, NJ, USA, 2012
- Master of Science in Internet Engineering,
  New Jersey Institute of Technology, Newark, NJ, USA, 2005
- Bachelor of Science and Master of Science in Electrical Engineering,
  Université de Lomé, Lomé, Togo, 2001

Major: Computer Engineering

Presentations and Publications:

K. Egoh, S. De, and R. Rojas-Cessa, "On the Selection of Optimal Relays in Wireless Multi-hop Packet Forwarding," *2012 (under review)*

K. Egoh, S. De, and R. Rojas-Cessa, "Energy Consumption Modeling and Duty-cycle Optimization for Low-Power Wireless Sensor Networks," *2012 (under review)*

R. Rojas-Cessa, K. M. Salehin, and K. Egoh, "Evaluation of Switching Performance of a Virtual Software Router," *in Proceedings of IEEE Sarnoff Symposium*, May 2012 (accepted for publication)

S. De and K. Egoh, "Multi-Criteria Optimization for Relaying in Multi-Hop Ad hoc Sensor Networks," U.S. Patent No. 7,872,977 (issued Jan. 18, 2011),

K. Egoh, R. Rojas-Cessa, and S. De, "Minimum Energy Multi-Criteria Relay Selection in Mobile Ad Hoc Networks," In H. Venkataraman, G. Muntean, *Green Mobile Devices and Networks: Energy Optimization and Scavenging Techniques*, CRC Press, Taylor and Francis Group, USA

R. Rojas-Cessa, K. M. Salehin, and K. Egoh, "Experimental Performance Evaluation of a Virtual Software Router," *Local & Metropolitan Area Networks (LANMAN)*, 2011 18th IEEE Workshop on, Oct. 2011, pp. 1-2

K. Egoh, R. Rojas-Cessa, and N. Ansari, "Distributed Diffusion-Based Mesh Algorithm for Distributed Mesh Construction in Wireless Ad Hoc and Sensor Networks," *in Proceedings of the 2010 IEEE International Conference on Communications (ICC 2010)*, Cape Town, South Africa, May 2010. pp. 1-5

S. De, K. Egoh, and G. Dosi, "A receiver initiated power control multiaccess protocol in wireless ad hoc networks", *in Proceedings of IEEE Sarnoff Symposium*, Princeton, NJ, May 2007. pp. 1-5.

K. Egoh and S. De, "A Multi-Criteria Receiver-Side Relay Election Approach in Wireless Ad Hoc Networks," *in Proceedings of the 2006 Military Communication Conference (MILCOM 2006)*, Washington, DC, Oct. 2006. pp. 1-7

K. Egoh and S. De, "Priority-based receiver-side relay election in wireless ad hoc sensors networks", *in Proceedings of Int. Conf. Wireless Communication Mobile Computing*, Vancouver, BC, Canada, Jul. 2006, pp. 11771182.

K. Egoh, "Stochastic Modeling of MAC Protocol Related Power Consumption in Multi-hop Wireless Ad hoc and Sensor Networks", *in Master Thesis*, Department of Electrical and Computer Engineering, NJIT, Newark, NJ, August 2005.

In science, 'fact' can only mean 'confirmed to such a degree that it would be perverse to withhold provisional assent.' –
Stephen Jay Gould

ACKNOWLEDGMENT

It would not have been possible to write this doctoral dissertation without the help and support of the kind people around me, to only some of whom it is possible to give particular mention here.

I would like to thank my advisor, Dr. Roberto Rojas-Cessa, for his support and great patience at all times. Dr. Rojas-Cessa puts great attention to the details, insists on scientific soundness, and provide encouragement when needed. I am grateful for his advise and mentorship. I would also like to express my gratitude to my co-adviser, Dr. Nirwan Ansari, and my former adviser, Dr. Swades De, who both have shown special interest in me, and have provided support, and guidance for many years.

I would like to thank Dr. Stewart D. Personick, Dr. Osvaldo Simeone, and Dr. Guiling Wang for accepting to serve as members of my dissertation committee. I am grateful for the time they have dedicated to this task and for all their questions and suggestions for improving the work.

I am blessed with a wonderful and supportive family who has been there for me and continues to be there for me. I owe them everything that I am. I thank my parents, my two brothers and three sisters for their love, prayers and unconditional support. I thank Gisele and Jaydon for putting up with my late-night studies and absence. I thank Marie-Lawoè and Simon Akakpo for their support throughout the years. I thank Messan Minyanou for his friendship and support.

Finally, this dissertation will not have been possible without the help and encouragement from countless friends and colleagues, many of whom I have come to consider my second family. To all of them, I am eternally grateful.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

Various packet forwarding schemes have been proposed for multi-hop wireless networks, where a transmitting node selects one of its neighbors to relay data packets toward the destination. In these schemes, a simple criterion, such as the relaying neighbor's geographical proximity to the final destination or the energy required to transmit a packet, is used by the transmitting node to select the best possible relay. Such forwarding approaches require that a list of all local neighbors be maintained at each node. However, maintaining a local neighborhood information at all nodes in a dense network with dynamic network environment and making sure the selected relaying node is active (e.g., by wake up signals or coordinated sleep patterns) may be costly for the resource constrained nodes [1–4].

The vast majority of the proposed rules for selecting the next forwarding neighbor assume a unit disk area coverage model. The widely used unit disk model assumes that a node within the coverage range is considered perfectly reachable, and nodes outside the range are unreachable [5–10]. Moreover, many available forwarding rule use only a single metric (e.g., one-hop progress, remaining energy) to choose the best candidate. However, in reality, the disk coverage assumption does not hold good from physical layer perspective (for example, see Figure 1.1), and a single criterion based forwarding node selection may not achieve the goal of network-wide optimal performance.

As an example, the hop-count based greedy geographic forwarding approach [11–32] has received a great deal of attention in the ad hoc networking research community. In this approach, by selecting the next 'best' relay in a greedy geographical faction, a transmitter

also tends to select relay nodes with poor link quality. For this reason, there have been a growing acceptance that the traditional greedy forwarding approach is not optimal in most practical settings where the disk coverage assumption or a *perfect-reception-within-range* does not hold [33]. To remedy this, a measure to the link quality between the transmitter and the potential relay node needs to be factored into the next hop relay decision. Although some link-aware routing schemes have been reported recently [33–35], the tradeoffs between greediness and link quality have not been thoroughly studied.

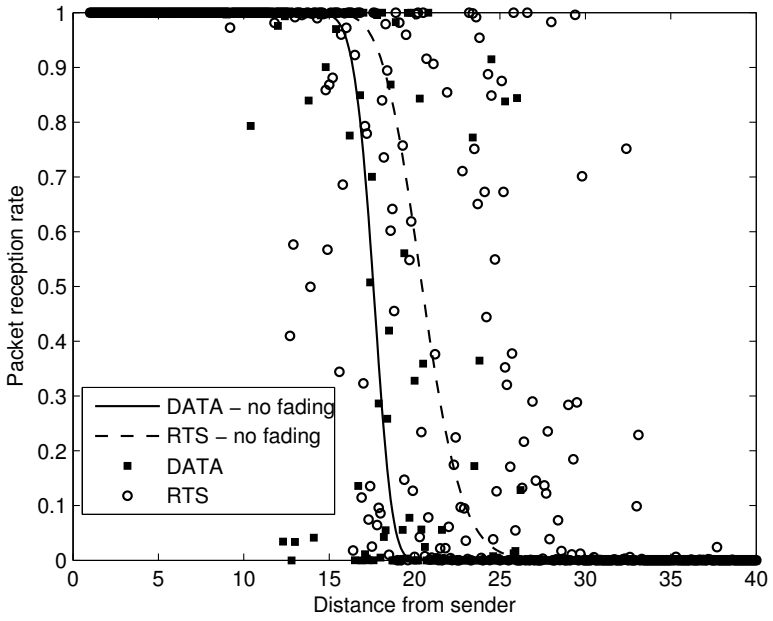


Figure 1.1 Sample of reachability in realistic wireless settings.

The challenge in considering more than one criterion(e.g. geographic location, link quality, delay, remaining energy) for the next hop relay decision lies in deciding the optimality of a particular relay candidate with respect to other nodes. Different criteria could have conflicting goals. Furthermore, the familiar scalar notion of optimality does not hold when multiple criteria are considered. Also, although the *transmitter-side relay selection* has the convenience of 'centralized' decision making, a transmitter has the

additional burden of gathering and maintaining the value of decision parameters for all relay candidates (neighboring node). This increases communication overhead and the risk of relay selection based on inaccurate or outdated decision information.

An alternative to transmitter-side relay selection is *receiver-side relay election*, in which the transmitting node does not decide of the next hop relaying neighbor. Rather, all neighbors contend among themselves to elect the best possible relay [10, 12, 36, 37]. With receiver-side relay election scheme, information such as received signal strength, remaining energy, are available at each potential relay node, and can be readily included in the decision making process. However, even in a single optimality criterion based forwarding (e.g., one-hop progress), the receiver-side relay election introduces additional challenge of vulnerability to collision because of the distributed nature of the election process.

This dissertation considers the problem of multi-criteria relay selection in multi-hop wireless networks, where in each hop among many candidate relays one is selected to relay data packets toward the destination. A generalized multi-parameter mapping function is introduced to aggregate all decision criteria into a single *aggregate criterion* used to rank the potential relay candidates. Optimal rules for the selection of the next hop relay are investigated as applicable to both transmitter-side selection and receiver-side election forwarding schemes. Beyond the theoretical formulation of the generalized multi-criteria based optimum election, as a demonstrative example of network performance evaluation, the network performance based on two optimality criteria is presented. Namely *forward progress* (greediness) and *packet success rate* (link quality) are used as decision criteria. It is shown that a suitable mapping function can be found which trades off the greediness for link quality and outperforms the reported transmitter-side link-aware forwarding schemes. Compared to the other schemes, the distributed two-criteria optimization results show a

substantially better end-to-end delay performance and a reduction of up to 5 times in end-to-end packet loss for the same required energy.

1.1 RELATED WORK

1.1.1 Optimal Relay Selection

Position-based forwarding, also called geographic routing, is a packet forwarding scheme that relies on geographic coordinate instead of network address. Using geographic coordinate data packets from a source station (also called netwoek node) are forwarded through multiple relay nodes toward the final destination. The question of *how* to select the next hop relay for optimum multi-hop communication has long been considered in packet radio networks [11, 13, 38, 39]. Typically, the node currently holding the data packet forwards the packet to a relay node selected among its neighbors based on a predefined selection rule or metric. Many proposed position-based forwarding solutions use metrics that are function of the locations of the final destination and the candidate relay node.

One of the first proposed rules for relay selection, named *most forward with r* (MFR) [38], uses geographic coordinate to select, within a radius of r, the neighbor which offers the most progress toward the final destination. The MFR forwarding strategy is the basis of the widely adopted greedy geographical forwarding [11, 12] where the goal is to achieve maximum progress with each transmission. Greedy forwarding is a good strategy in a network with homogeneous nodes (i.e. same transmission range and without transmit power control) [39]. Reachability in these networks is often modeled by an idealized disk within which nodes are assumed perfectly reachable. While greedy forwarding ensure minimum number of hops between source and destination, each hop however may require many re-transmissions. In most practical wireless networks with unreliable wireless link,

higher retransmission rates leads to diminished throughput, higher delay and higher energy consumption.

Many attempts have been made to remedy the limitation of greedy forwarding. In [40], *nearest with forward progress* (NFP) improves throughput by selecting a relay candidate closer to the transmitter in order to improve throughput. A few link-aware metrics which combine geographic location with some measure of link quality improve on the performance of the traditional greedy forwarding [33] [34]. By incorporating a measure of link quality into the selection metric, link-aware forwarding schemes are able to avoid poor links candidates by prioritizing good link. The link-aware metrics however suffer additional overhead when implemented as transmitter-side selection because of added link quality information gathering (or estimation) requirement.

1.1.2 Link-aware Geographic Routing

In some recent performance studies on ad hoc networks, link quality has been taken into account along with the progress toward the destination to choose an optimum forwarding node [33,34,41]. In [33], a simple product form of packet success probability and progress toward the destination was considered, and an optimum node was selected that offers the maximum value of the product. The selection of next hop in [34] was based on a normalized advance (NADV) using different link costs (e.g., packet error rate, energy, delay) as normalizing factor. When packet error rate is considered along with advancement toward the destination, NADV is also equivalent to the product form as in [34]. Likewise, in [35] the same cost metric was applied in distance dependent loss aware geographic multi-hop relaying.

In chapter 3 it is shows that the simple product form (*PROD*) can be outperformed by an optimum tradeoff between greediness and link quality. Additionally, the multi-criteria relaying framework can potentially accommodate any number of constraints in selecting/electing a next-hop node. As a specific two criteria case example is considered, with geographic location and link quality, and it is shows that a judicious selection of weights of these 2 criteria can significantly improved network throughput and energy consumption performance.

1.1.3 Opportunistic Forwarding and Distributed Schemes

Until recently, all position-related forwarding schemes proposed to make the selection of the next hop node at transmitter-side. These schemes may work well with lightly populated, and relatively static ad hoc networks. However more dynamic, dense, and resource constrained networks, such as sensor networks, require that the question of *where* to make the next hop selection be reconsidered.

Another group of proposals aimed at improving the performance of greedy forwarding select the next relay node only among neighbors which have successfully received the data packet (or an initiating system packet) from the transmitting node. These proposals, generally termed *opportunistic*, also offer a distributed alternative to the centralized (i.e. transmitter-side) relay selection. In [12] and [36], authors have independently considered forwarding schemes in which transmitting nodes do not need to select the next hop. Rather, all eligible candidates compete among themselves to relay the packets. While [12] considered remaining distance to the destination based forwarding node selection priority criteria, it did not capture the additional MAC contention in the selection process. Rather, it was assumed on the one hand that somehow the best relay

is always elected and on the other hand that the selection process is always successful. The authors of [36] studied three possible variants of forwarding node selection aiming at reduced packet duplication, where it was assumed that more than one nearly-simultaneous responses could be successful. Priority dependent MAC contention probability and the related delay in successful relay selection process was not considered. In [42], an opportunistic carrier sensing scheme is considered with a time backoff based distributed scheduling used to prioritize nodes. The author presents a formulation of the distributed scheduling as mapping of decision criterion onto the scheduled time backoff. In the distributed schemes [12, 36, 42], the transmitting node does not decide about selecting the next hop relaying neighbor. Instead, all neighbors contend among themselves to elect the best possible relay. Nevertheless, these distributed methods still rely on a single location-based decision criterion and face performance degradation and higher energy consumption in lossy wireless environments.

For opportunistic and distributed-forwarding schemes to work certain networking conditions need to be ensured. Specifically, every node desiring to transmit should find with high probability at least one active neighbor in the forward direction to relay its packets. The feasibility and stability of such networks conditions have received some attention in the research community. [43] considered a network of independent Bernoulli type nodes and derived the limiting probability that every node has at least one active neighbor in highly dense networks. [44] also considered the feasibility of a network of sensor nodes with independent asynchronous duty-cycles in which transmitting nodes can simply broadcast their packets and have them relayed by available active neighbors. However, because the main focus of the authors is robustness, the approach does not discourage packet duplication and may not be optimal in dense networks.

1.1.4 Meshed Multi-Path Forwarding

In [45], the authors of AODV-MM used the mesh type shown in Figure 4.5(b) to demonstrate the merits of their proposed mesh multipath routing protocol. By using a combination of route discovery request signals and overhearing, each node is able to maintain a primary route and a series of secondary routes to the destination. In DDM the same mesh is constructed by using rule 10 and an initial token vector of $[1\ n\ 1\ 0]$, where n represents the number of hops separating the source S from the destination D. All comparisons with AODV-MM were made using this mesh.

CHAPTER 2

RECEIVER-SIDE RELAY ELECTION

2.1 Motivation

In multihop wireless networks, the choice of intermediary relay node at each hop has been traditionally made by a transmitting node, which selects one of its neighbors to relay data packets toward the destination [11, 39]. In these schemes, a single criterion such as the relaying neighbor's geographical proximity to the transmitter or the final destination, or the energy required to transmit a packet, is used by a transmitting node to select the best possible relaying neighbor. Such forwarding approaches require that a list of all local neighbors be maintained at a node. However, maintaining a local neighborhood list at all nodes in a dense network with dynamic network environment and making sure the selected relaying node is active (e.g., by wake up signals or coordinated sleep patterns) may be costly for the resource constrained sensor nodes.

An alternative to transmitter-side relay selection is a forwarding scheme in which the transmitting nodes do not decide which one of their neighbors would relay their data packets, rather all relay candidates contend among themselves to relay the packets toward the final destination [12, 36]. Ideally, to avoid packet duplication, a single neighbor should be elected among all candidates as next hop relay. This contention resolution process will be referred to as *receiver-side relay election*. One essential advantage of receiver-side relay election is that complex decision criteria such as received signal strength or residual energy at candidate relays can be considered in a more efficient way which would have been infeasible or too costly in case of transmitter side relay selection.

Note that, because the relay election process is a distributed decision, multi-access collisions are likely to occur among two or more best relay candidates. When a collision happens, the election process fails, increasing the delay incurred to data packets. The effectiveness of receiver-side relay election process is best characterized firstly by the quality of the elected relay with respect to a chosen decision criterion, and secondly by the level of vulnerability of the election process to collisions. To elect a *good* relay node at the end of the contention resolution period, different decision criterion, such as, maximizing per-hop progress toward the destination, minimizing per-node energy consumption, maximizing the end-to-end throughput, etc., can be used as a basis of prioritization among potential relays. In [12], one hop progress toward the final destination is used as the priority criteria. However, the authors considered no specific contention scheme in their routing performance analysis.

It is observed that, while the receiver-side relay election approach could be a potential low-cost and distributed solution to energy optimized multihop wireless routing, a careful investigation on medium access control (MAC) contention resolution and the network performance is necessary to assess the end-to-end performance more accurately.

In this section, the MAC contention issues in different priority-based receiver-side relay election schemes and the associated per-hop delay performance are analysed. The focus is on the performance of two main schemes, namely, *random forwarding* and *priority forwarding*, where the priority criteria is chosen based on the forwarding node's distance to the destination. The proposed scheme introduces a generalized mapping function for the prioritization, in which the shape of relative scheduled time priority is governed by a mapping shape parameter. Through probabilistic analysis, supported by simulations, it is shown that, in a given network setting, there exists a suitable shape parameter that offers

an optimum relay election performance in terms of per-hop progress and effective delay in successful relay election process. The results could be used to introduce more complex decision criteria in relay election process, such as energy-awareness, power control, and mobility, to obtain an optimal network wide performance.

The rest of this chapter is organized as follows. The basics of receiver-side relay election and formal definitions of the prioritization schemes are presented in Section 2.2. The analytic framework for performance evaluation of the prioritization schemes is presented in Section 2.3. Analytical and simulation results are contained in Section 2.4. Concluding remarks are drawn in Section 2.5.

2.2 Receiver-Side Relay Election

Consider a network of uniformly random distributed nodes with homogeneous and circular coverage, and independent and asynchronous sleeping behavior. It is assumed, similar to 802.11 distributed coordination function (DCF), RTS/CTS (request-to-send/clear-to-send) message exchange is done between the transmitter and a potential forwarder before the data packet forwarding. However, unlike in 802.11, the RTS message is broadcast to all local neighbors, and a forwarder's CTS response is suitably delayed to minimize the potential contention. It is also assumed that a node is aware of its geographic location or virtual (hop-count based [46]) location information of its own and the destination. The following subsections present a further elaboration of the concepts of contention resolution, vulnerability to collision, and prioritization strategies in relation to receiver-side relay election.

2.2.1 Contention Resolution

Hop-by-hop forwarding is generally done in packet radio networks based on a decision criteria to obtain desired nodal or network-wide performance objective. In receiver-side relay election, these decision criteria are (implicitly) used as priority measures in the distributed relay election. Before the analysis of the performance of various prioritization schemes, it is helpful to present the description of a basic election process using time delay as contention resolution.

A node desiring to send data packet first sends a broadcast RTS packet containing the optimality-criteria and location information of itself and the final destination. After receiving the RTS packet, every eligible relay candidate i (the shaded region in Figure 2.1) schedules a reply time:

$$X_i = g(\Omega_i), \tag{2.1}$$

where Ω_i is the quality measure of node i computed based on a given criterion used by the forwarding scheme. $g(\cdot)$ is a mapping function that implements the prioritization of the election process and its nature determines the quality of the elected relaying neighbor with respect to the set of optimality criteria and the vulnerability of the election process to collision among two or more of the best candidates.

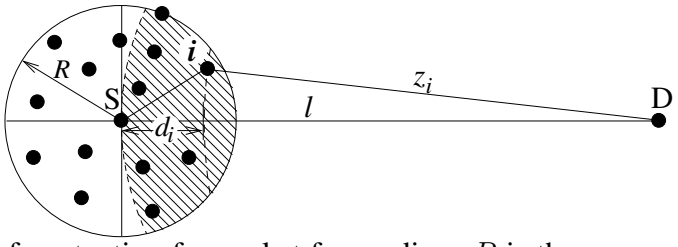


Figure 2.1 Area of contention for packet forwarding. R is the coverage range, and i is a forwarding contender, which offers forward progress d_i towards the final destination D.

Next, every relay candidate i listens to the wireless medium between the time 0 and X_i. If no other CTS is received before time X_i, then node i considers itself the winer of the election process and sends a CTS packet with its signature to the transmitting node. If a node overhears a CTS transmission during its waiting period, it gives up the contention, assuming that a better forwarding candidate has been found.

2.2.2 Vulnerability to Collision

Because of the distributed nature, receiver-side relay election processes are vulnerable to collision. The election process fails to elect single next hop when two or more relay candidates schedule the same or very close RTS reply times. Note that, this type of collision is different from (and additive to) regular medium access collision such as those caused by hidden or exposed terminals. Since all forwarding schemes, independently of whether the relay selection is done at the transmitter-side or at the receiver-side, are subject to the same regular medium collision, in this work the focus is not in quantifying this type of collision.

To quantify the collision in the election process, assume node j schedules reply time $X_j = \min_i \{X_i\}$. A collision occurs if there exist a at least one node k ($k \neq j$) such that

$$|X_j - X_k| \leq \beta, \tag{2.2}$$

where β is the collision vulnerability window, which may depend on the MAC scheme, nodes' clock precision, signal detection time, and receive-transmit changeover delay.

Upon correctly receiving a CTS packet, the transmitter sends the data packet to the forwarder. If the transmitter receives another correct CTS packet afterwards for the

same data packet (if the earlier CTS packet was unheard by some nodes in the forwarding zone), it simply discards the CTS to avoid any packet duplication. In case of any CTS collision, all forwarding nodes give up in that contention cycle, and, as in 802.11 DCF MAC contention resolution. It is assumed that the transmitter re-initiates the election process by broadcasting another RTS packet after a timeout.

2.2.3 Prioritization Types

Prioritization functions can be categorized into two main groups: *purely random* and *absolute priority-based*. The following definitions of the two prioritization types assume, without loss of generality, that the desired objective is to maximize the decision criteria Ω. For the sake of completeness, a variation of forwarding scheme, called *hybrid priority*, is also defined, which is derived from the combination of the priority-based approach and the random approach.

Definition 1 *In **purely random** forwarding, no priority is given to better candidates during the election process. If nodes j and k are two relay candidates, then the fact that node k is a better candidate than node j ($\Omega_j \leq \Omega_k$) does not increase the chances that k is elected over j. Formally,*

$$Pr[X_k \leq X_j | \Omega_j \leq \Omega_k] = Pr[X_k \leq X_j]$$

Note that in this case the decision criteria Ω does not have any role in electing a relay.

Definition 2 *In **absolute priority** forwarding, absolute priority is given to better candidates during the election process. If nodes j and k are two relay candidates, then the fact that k*

is a better candidate than j ($\Omega_j \leq \Omega_k$) guarantees that k is elected over j. That is,

$$Pr[X_k \leq X_j | \Omega_j \leq \Omega_k] = 1$$

Note that absolute priority can be obtained with a function $g(\cdot)$ which is deterministic and monotonically decreasing with respect to Ω.

Definition 3 *A **hybrid priority** combines absolute priority and purely random selections, and gives priority but no guarantee to a better candidate in the election process. If nodes j and k are two relay candidates, then the fact k is a better candidate than j ($\Omega_j \leq \Omega_k$) increases the chances but does not guarantee that k is elected over j. That is,*

$$Pr[X_k \leq X_j] < Pr[X_k \leq X_j | \Omega_j \leq \Omega_k] < 1$$

However, hybrid priority will not be discussed further in this paper; it will be left as a future research extension.

2.3 Analysis of Prioritization Schemes

The section determines the characteristics of the prioritization function $g(\cdot)$ (see (2.1)) and analyses the priority-specific performance of relay election processes. Particular interested is given to characterizing the vulnerability to collisions and the effective delay in successful relay election process.

It is assumed that based on the location information and a set priority criteria in the RTS packet each node can have its own measure Ω_i of forwarding decision.

Two main time delay based contention resolution criteria that are considered are random delay and prioritized delay. In case of random delay, a node in the forwarding zone picks up a random waiting time drawn from a given probability distribution function (pdf) before sending its CTS message. Uniformly random ($\texttt{uni\_rand}(t_2, t_1)$) distribution are considered for the chosen waiting time in the range $[t_2, t_1]$. The decision criteria Ω in this case is independent of the nodes locations (as long as they are in the forwarding region (cf. Figure 2.1)), and thus the waiting time X_i of node i, i.e., the mapping function $g(\cdot)$, is the corresponding random distribution itself.

On prioritized election process, there could be many possible criteria, namely, maximum residual energy based, receiver signal strength based, maximum per-hop progress based, etc., or a combination of them. In this study, the maximum per-hop progress toward the destination is chosen as the absolute priority criteria. Accordingly, Ω_i represents the approaching progress toward the destination d_i a relay candidate i can offer, if elected (cf. Figure 2.1). In this case, the mapping function $g(\Omega_i)$, which is monotonically decreasing with respect to d_i, can be expressed as

$$X_i = g(\Omega_i) = \{a(\alpha)d_i + b(\alpha)\}^{1/\alpha} \tag{2.3}$$

where the coefficients $a(\alpha)$ and $b(\alpha)$ are chosen such that the prioritized delay of a forwarding region node remains within $[t_2, t_1]$, as in the $\texttt{uni\_rand}$ case. α is the shape parameter ($\alpha \neq 0$) that governs the nature of relative priority of potential relay nodes. $a(\alpha)$ and $b(\alpha)$ are given by:

$$a(\alpha) = \frac{t_2^\alpha - t_1^\alpha}{R}; \quad b(\alpha) = t_1^\alpha \tag{2.4}$$

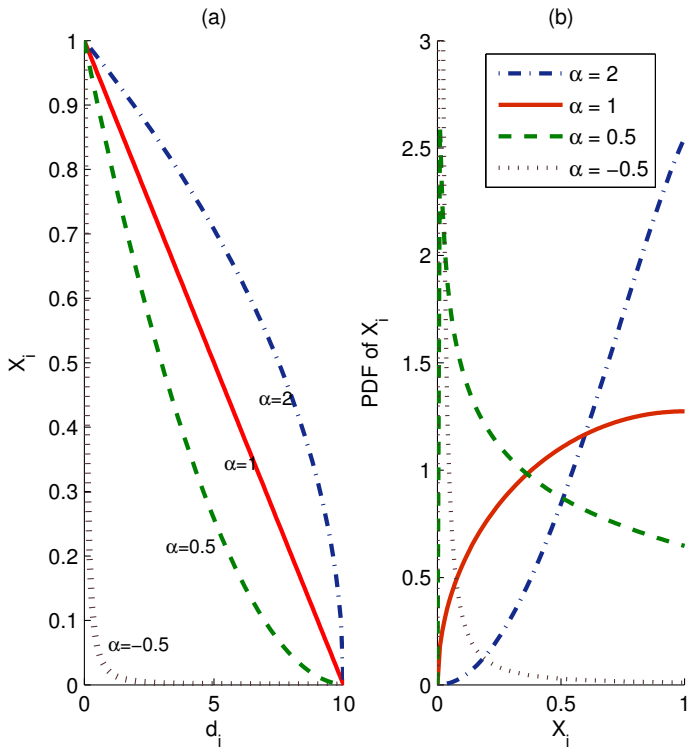


Figure 2.2 (a) Mapping functions in one-hop progress based absolute priority as a function of α; (b) corresponding pdf's of mapped random variable X_i. $R = 10$.

In Figure 2.2 the priority-based mapping function $g(\cdot)$ and the corresponding mapped random variable X_i (scheduled time of node i) as a function of the shape parameter α is depicted. It is interesting to observe that, by varying α, relative priority of different node election can be achieved. For example, $\alpha = 1$ corresponds to the linear mapping with respect to one-hop progress, whereas $\alpha = -1$ corresponds to the inverse mapping. It is remained to be seen the effect of different mapping on the relay election delay and collision vulnerability, which will be discussed subsequently.

In the following development, first the generic expressions for delay and collision probability are derived. Specific cases of relay election strategies are taken up as examples.

2.3.1 Relay Election Delay

To find the time required to successfully elect a relay, denote by $\{X_i\}_{i=1,2,\cdots}$ the set of independent and identically distributed random variables representing the scheduled reply times of all eligible relay candidates. The number of such contenders is itself another random variable C, which is Poisson distributed with parameter λ, the average number of active forward direction neighbors. The conditional duration of one election process is define as $Y = \min_i \{X_i\}$. The interest is in the distribution ($f_Y(y)$ and $F_Y(y)$) of Y for an arbitrary prioritization scheme, i.e., for arbitrary distribution ($f_X(x)$ and $F_X(x)$) of the X_i's.

If $C = c$ active contending candidates are considered, each with a scheduled time X_i $\{i = 1, 2, \cdots, c\}$, the conditional cumulative distribution function (cdf) of Y is obtained as

$$\begin{aligned}
\Pr[Y \leq y | C = c] &= 1 - \Pr[Y > y | C = c] \\
&= 1 - \Pr[\min_i \{X_i\} > y] \\
&= 1 - \prod_{i=1}^{c} \Pr[X_i > y] \\
&= 1 - \{1 - F_x(y)\}^c
\end{aligned}$$

Note that Y is defined only for $c \geq 1$. The unconditional cdf of Y can then be obtained by total probability:

$$\begin{aligned} \Pr[Y \leq y] &= \sum_{c=1}^{\infty} \Pr[Y \leq y | C = c] \frac{\Pr[C = c]}{\Pr[c \geq 1]} \\ &= \sum_{c=1}^{\infty} \left[1 - \{1 - F_x(y)\}^c\right] \frac{\frac{\lambda^c}{c!} e^{-\lambda}}{1 - e^{-\lambda}} \end{aligned}$$

After simplification, the total probability becomes

$$\begin{aligned} F_Y(y) &= \Pr[Y \leq y] = \tfrac{1 - e^{-\lambda F_X(y)}}{1 - e^{-\lambda}} \\ f_Y(y) &= \tfrac{\lambda f_X(y) e^{-\lambda F_X(y)}}{1 - e^{-\lambda}} \end{aligned} \tag{2.5}$$

Below, the general result of equation (3.3) is applied to two examples of prioritization.

Uniformly Random Forwarding Any forwarding solution in which all relay candidates contend by setting a purely random timer falls into this category. As an illustration, the uniformly distributed random time between t_2 and t_1 is considered, i.e.,

$$\begin{aligned} f_X^{rand}(x) &= \tfrac{1}{t_2 - t_1} \\ F_X^{rand}(x) &= \tfrac{x - t_2}{t_2 - t_1} \end{aligned}$$

Then, closed form expressions for the distributions with uniformly random X_i can be obtained from equation (3.3):

$$\begin{aligned} F_Y^{rand}(y) &= \frac{1-e^{-\lambda\frac{y-t_2}{t_1-t_2}}}{1-e^{-\lambda}} \\ f_Y^{rand}(y) &= \frac{\lambda}{(t_1-t_2)\left(1-e^{-\lambda}\right)} e^{-\lambda\frac{y-t_2}{t_1-t_2}}. \end{aligned} \tag{2.6}$$

Absolute Priority with Linear Mapping For the absolute priority forwarding, the case where scheduled time is a function of one-hop progress and linearly decreasing between t_1 and t_2 is considered. Then, from equations (3.1)-(3.2), setting $\alpha = 1$, $X_i = \frac{t_2-t_1}{R} d_i + t_1, 0 \leq d_i \leq R$, where d_i is the forward progress to the destination of candidate relay i (cf. Figure 2.1). The distribution of X_i can be deduced from the distribution of remaining distance [13], and can be approximately given by

$$f_{d_i}(d) = \frac{4(l-d)}{\pi R^2} \arccos\left(\frac{(l-d)^2 + l^2 - R^2}{2l(l-d)}\right) \tag{2.7}$$

The distributions of the scheduled time are given by

$$\begin{aligned} F_X^{lin}(x) &= 1 - F_{d_i}\left(\frac{t_1-x}{t_1-t_2}R\right) \\ f_X^{lin}(x) &= \frac{1}{|t_1-t_2|} f_{d_i}\left(\frac{t_1-x}{t_1-t_2}R\right) \end{aligned} \tag{2.8}$$

From equations (2.6) and (2.8) the average delay in one relay election attempt can be obtained in these two cases. Delay results for these two examples along with the other variants will be presented in Section 2.4.

2.3.2 Election Failure Probability

As stated earlier (see equation (2.2)), the election process fails when the first two best candidates are not at least the collision vulnerability window apart from each other. Formally, denote Y as the minimum of the set of scheduled reply times: $Y = \min_i \{X_i\}$, and Y^* as the minimum of the remaining remaining nodes' scheduled reply times: $Y^* = \min\{\{X_i\} - Y\}$. Note that Y and Y^* can be considered identically distributed. Also, define $S_Y(y) = 1 - F_Y(y)$ as the survival function of Y, and $h(y) = \frac{f_Y(y)}{S_Y(y)}$.

Lemma 1 For a given collision vulnerability window β, the rate of failure of the election process is given by

$$P_{fail} = 1 - (h \copyright S_Y)(\beta) \tag{2.9}$$

where © represents the correlation integral function defined by

$$(h \copyright S_Y)(t) = \int_{-\infty}^{\infty} h(x) S_Y(t + x) dx$$

Proof: Under the condition $Y = y$, a collision occurs with the conditional probability $\Pr[Y^* \leq y + \beta | Y = y] = \frac{F_Y(y+\beta) - F_Y(y)}{1 - F_Y(y)}$. The unconditional probability of collision can

then be obtained by

$$\begin{aligned} P_{fail} &= \int_{t_2}^{t_1} \Pr[y \leq Y \leq y + dy] \frac{F_Y(y+\beta) - F_Y(y)}{1 - F_Y(y)} \\ &= \int_{t_2}^{t_1} f_Y(y) dy \frac{F_Y(y+\beta) - F_Y(y)}{1 - F_Y(y)} \\ &= \int_{t_2}^{t_1} f_Y(y) \frac{S_Y(y) - S_Y(y+\beta)}{S_Y(y)} dy \end{aligned}$$

After simplification, realizing that for $y \geq t_1,\ S_Y(y) = 0$, and for $y \leq t_2,\ h(y) = 0$,

$$P_{fail} = 1 - \int_{t_2}^{t_1} h(y) S_Y(y+\beta) dy$$

and hence the proof. □

Failure probabilities for specific prioritization cases can be obtained from the respective distribution functions.

2.3.3 Effective Delay

Because the distributed relay election process can fail due to collision, repeated attempts may be required for successfully finding a next hop. To have a baseline comparison of priority-specific approaches, one needs to compute the effective delay of an eventually successful relay election process. For this purpose, and for simplicity, it is assumed that in case of a collision of a RTS reply, the sender re-initiates another election process at the end of a fixed timeout window (t_1), and repeats this process until a CTS is received

successfully.[1] The effective average delay before a relay is successfully found is given by

$$D_{eff} = \frac{P_{fail}}{1 - P_{fail}} t_1 + D, \tag{2.10}$$

where P_{fail} is the collision probability given in equation (3.4), and D the average delay is one attempt, obtained from equation (3.3).

2.4 Results

To evaluate the prioritization performance in receiver-side relay election process, consider $\texttt{uni\_rand}(t_2, t_1)$ case and least remaining distance to the destination (LRD) based [13] priority forwarding forwarding, with the mapping function given in equation (3.1). The range of waiting time in both cases are set in $[t_2, t_1]$. The mapping parameter α is varied to give different priority levels to different eligible nodes.

Nodes are assumed to have independent and asynchronous sleep behavior, leading to a different realization of set of active nodes eligible for forwarding in each transmission attempt by a sender. It is also assumed that the average number of active neighbors n of any node (also referred as node density) is stationary. For simplicity in obtaining the analytic results, average number λ of relay candidates of a node, denoted by the shaded region in Figure 2.1, is approximated as $\lambda = \frac{n}{2}$.

All numerical and simulation results in this section are obtained with node transmission range $R = 10$ and a sink node at a distance $l = 100$ from the initial transmitting node. The collision vulnerability window is taken to be the changeover time

[1] In practice, each transmission failure is followed by a binary exponential backoff, and only a finite number of re-attempts are done.

of commonly available hardware ($\beta = 250\mu s$). The scheduled reply times range from $t_2 = 250\mu s$ to $t_1 = 1s$.

First, the delay in one relay election attempt for different priority is obtained with particularly interested given to the behavior of the relative priorities controlled by the shape parameter α. Figure 2.3 shows that the smaller α, the lesser the delay. This is

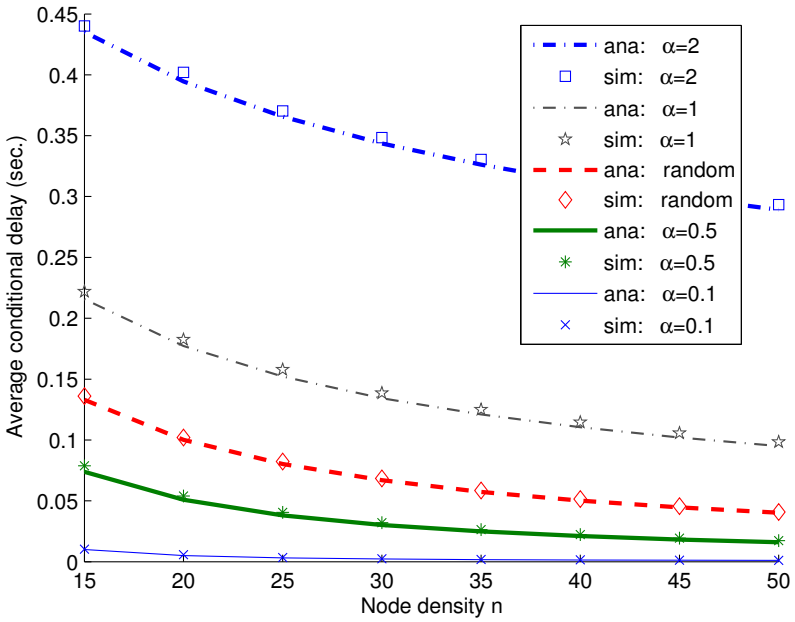


Figure 2.3 Delay vs. node density in one relay election attempt in different priority approaches.

because, as demonstrated in Figure 2.2(b), for lower value of α (< 1) the the scheduled time distribution is squeezed toward the lower values. On the other hand, $\alpha > 1$ stretches the distribution in the opposite direction, resulting in higher conditional delays.

The election failure probability P_{fail} plots in Figure 2.4 show a reverse trend as α changes. This is because at lower α, since the distribution of scheduled time is shifted toward the lower range, there is a high chance that two best nodes have nearly the same scheduled time (see equation (2.2)) for the CTS message, leading to a collision.

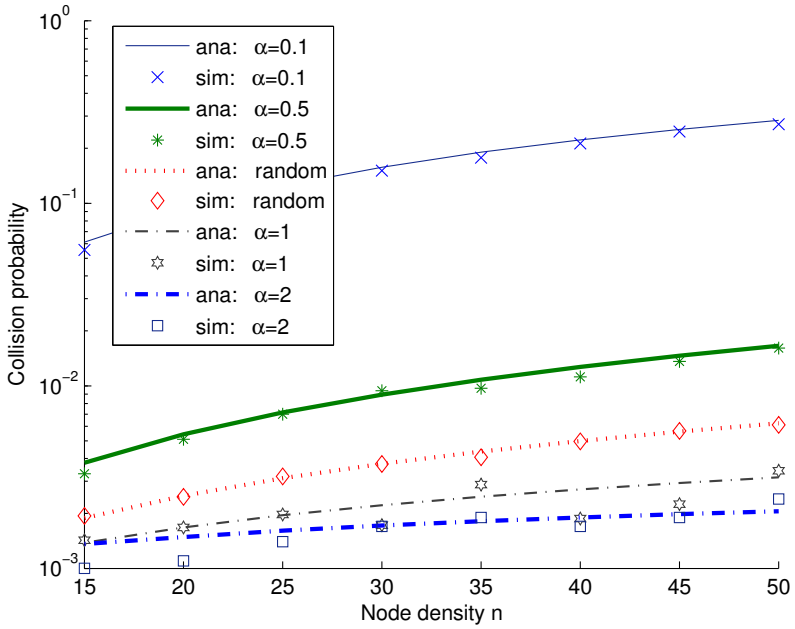


Figure 2.4 Election failure probability P_{fail} vs. node density n in different priority approaches.

With the observation of counter-directional trends of delay in one election attempt and election failure probability for different shape parameters, the effective delay performance in a successful relay election process can be obtained. Figure 2.5 shows the effect of the shape parameter on the effective delay. It is observed that an optimum shape parameter can be obtained to achieve the best tradeoff between delay and collision probability, which is also a slowly varying function of the node density. For example, at node density $n = 20$, the optimum shape parameter $\alpha_{opt} = 0.3$ and the effective delay is nearly 37.9 ms, whereas, at $n = 20$, $\alpha_{opt} = 0.5$ and the effective delay is nearly 31.5 ms.

Next, the effective delay in successfully electing a relay for different prioritization schemes is obtained. The intuition from Figure 2.5 is verified in Figure 2.6 via analysis and simulation that there is a critical value of shape parameter for which an optimum election performance can be achieved. It is also observed that unless the prioritization scheme is suitably optimized, its performance can be even poorer than the random election process.

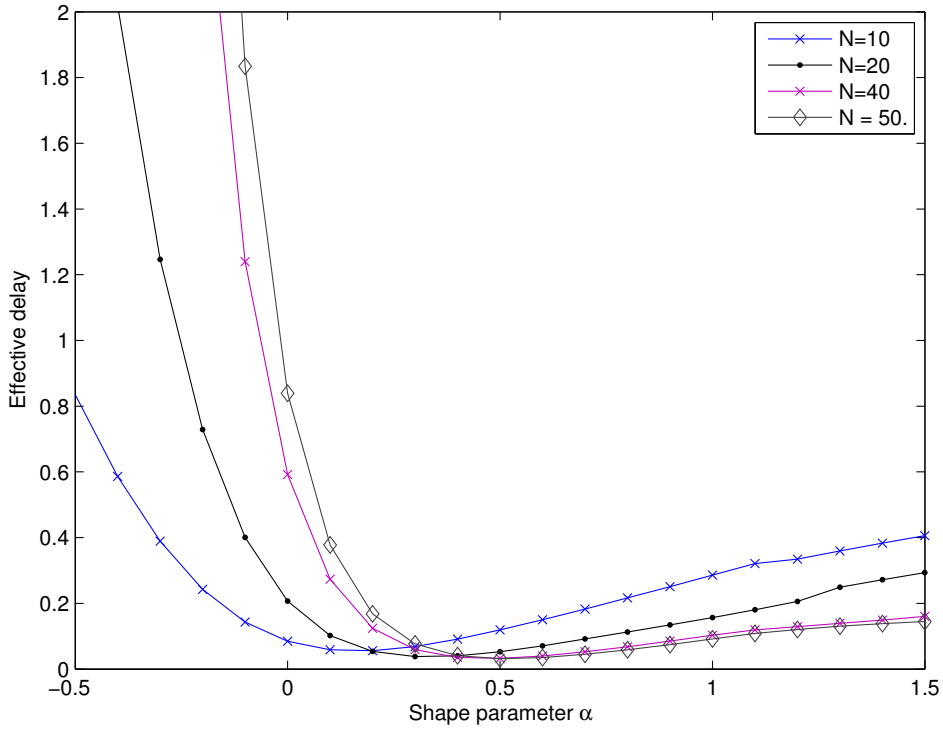


Figure 2.5 Analytical effective delay vs. shape parameter α.

Node that the average one-hop progress in LRD based absolute priority forwarding is independent of the shape parameter as the best relay is elected in all cases. It is observed that the average progress in random forwarding is nearly less than half of that in LRD approach (see Figure 2.7). Although random forwarding has an acceptable delay performance, which can be even better than the LRD based priority unless the shape parameter is chosen judiciously, a very poor forward progress makes the random forwarding a rather less promising approach.

2.5 Conclusions

A priority-based receiver-side relay election schemes is presented, where random forwarding and least remaining distance to the destination (LRD) based absolute priority

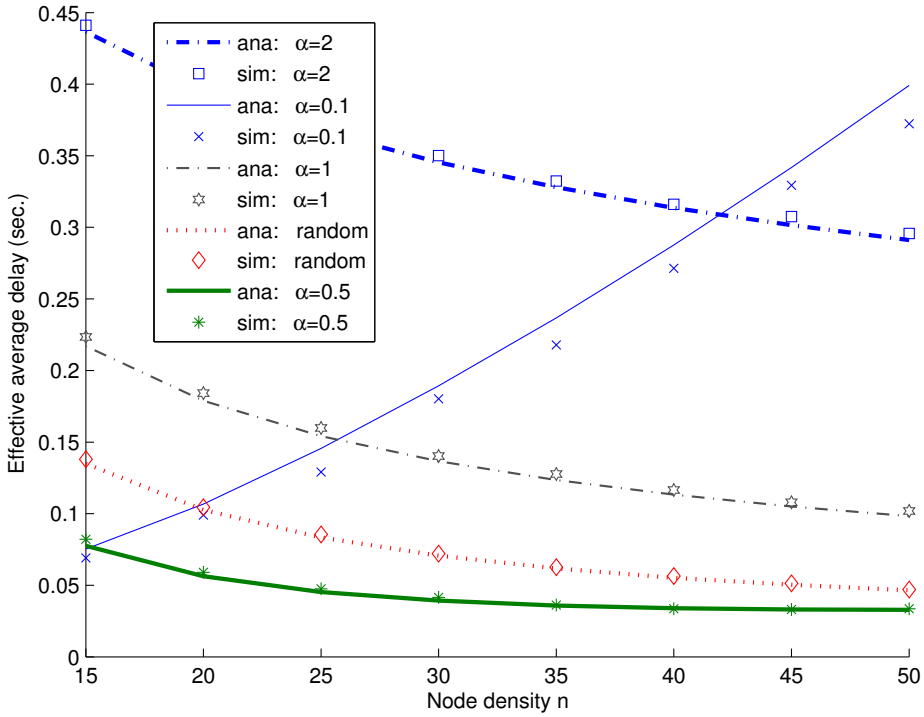


Figure 2.6 Effective delay in successfully electing a relay vs. node density in different priority cases.

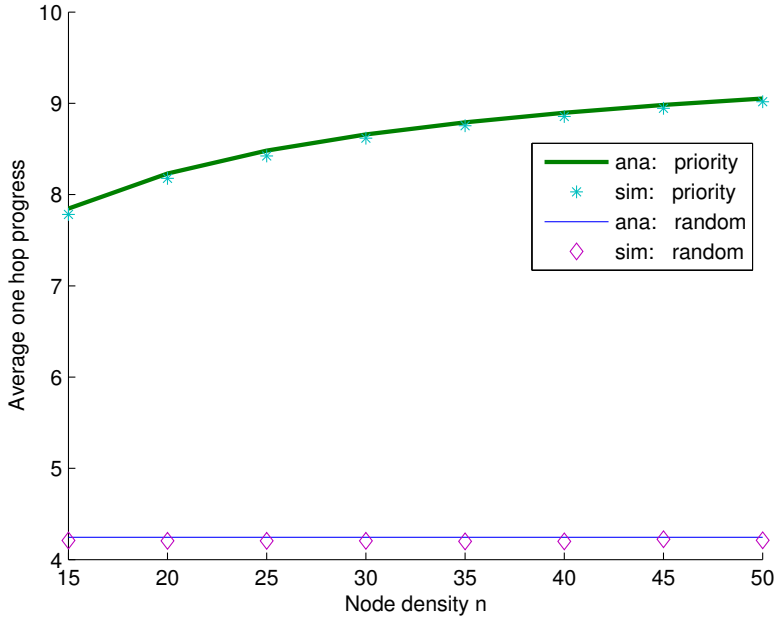


Figure 2.7 One-hop progress in random forwarding and LRD priority based approaches vs. node density, for any shape parameter α.

were considered. A generalized mapping function for LRD based priority was proposed, where the mapping shape parameter can control the relative priority of eligible nodes as desired. Via probabilistic analysis, supported by network simulations, performance of different priority schemes were compared in terms of delay in one election attempt, election failure probability, and eventually, effective delay in a successful relay election process.

While random forwarding is simple and offers reasonably good delay performance, its drawback is poor one-hop progress. In case of LRD based priority, the shape parameter controls the delay and collision probability. Although a higher value of shape parameter offers lower delay in one election attempt, it has higher failure probability. It was shown that for a given network parameter, an optimum mapping function can be obtained (with an appropriate shape parameter) that offers the best tradeoff between delay and collision probability, which is also superior to the random forwarding with respect to effective delay as well as one-hop progress.

CHAPTER 3

MULTI-CRITERIA RECEIVER-SIDE RELAY ELECTION

Traditional purely greedy forwarding in wireless ad hoc networks is not optimal in most practical settings where *perfect-reception-within-range* cannot be assumed. Although a few link-aware routing schemes have been reported, the tradeoffs between greediness and link quality have not been studied.

In this chapter, a multi-criteria based receiver-side relay election approach is taken in wireless multi-hop forwarding, where a single optimal node is elected among many candidates to relay packets toward the final destination. A general cost metric is introduced in the form of a multi-parameter mapping function, that aggregates all decision criteria into a single *virtual criterion* to rank potential relay candidates. It is shown that a suitable mapping function can be found, which trades off greediness for link quality to obtain optimal end-to-end network performance. Compared with the previously reported link-aware forwarding schemes, the results show a better energy performance and a substantial improvement in end-to-end delay.

The challenge in considering more than one criteria (such as link quality, delay, remaining energy) for the next hop selection lies in deciding the optimality of a particular neighbor with respect to other nodes, because different criteria could have possibly conflicting goals. In other words, the familiar scalar notion of optimality does not hold when multiple criteria are considered. Furthermore, although the *transmitter-side relay selection* has the convenience of 'centralized' decision making, a transmitter has the additional burden of gathering and maintaining all decision information.

An alternative to transmitter-side relay selection is *receiver-side relay election*, in which the transmitting node does not decide of the next hop relaying neighbor. Rather, all neighbors contend among themselves to elect the best possible relay [10, 12, 36]. With receiver-side relay election scheme, information on priority criteria, such as received signal strength, remaining energy, are readily available at each potential relay node, which can be easily included in deciding the next hop node. However, even in a single optimality criterion based forwarding (e.g., one-hop progress), the receiver-side relay election introduces additional challenge of vulnerability to collision because of the distributed nature of the election process.

In this chapter, the problem of multi-criteria receiver-side relay election in multi-hop wireless networks is considered, where among many candidate relays one is elected to forward packets toward the final destination. A generalized multi-parameter mapping function is introduced to aggregates all decision criteria into a single metric used to rank the potential relay candidates. Optimal rules for next hop relay selection, as applicable to both transmitter-side selection and receiver-side election, are investigated. Beyond the theoretical formulation of the generalized multi-criteria based optimum election, a demonstrative is presented with two decision criteria, namely forward progress (greediness) and packet success rate (link quality). It is shows that a suitable mapping function can be found which trades off greediness for link quality and outperforms the reported transmitter-side link-aware forwarding schemes. Compared to the other schemes, the distributed two-criteria optimization results show a substantially better end-to-end delay performance and a reduction of up to 5 times in end-to-end packet loss for the same required energy.

In some recent performance studies on ad hoc networks, link quality has been taken into account along with the progress toward the destination to choose an optimum forwarding node. In [33], a simple product form of packet success probability and progress toward the destination was considered, and an optimum node was selected that offers the maximum value of the product. The selection of next hop in [34] was based on a normalized advance (NADV) using different link costs (packet error rate, energy, delay) as normalizing factor. When packet error rate is considered along with progress to the destination, NADV is also equivalent to the product form as in [34]. Likewise, in [35] the same cost metric was applied in distance dependent loss aware geographic multi-hop relaying.

This chapter shows that the simple product form (*PROD*) can be outperformed by an optimum tradeoff between greediness and link quality. Additionally, the multi-criteria relaying framework can potentially accommodate any number of constraints in selecting/electing a next-hop node. For illustration, the two-parameter with hop-progress (greediness) and reachability (link quality) shows significantly improved network performance and nodal energy saving can be achieved.

The rest of the chapter is presented as follows. In Section 3.1, the basic receiver-side relay election approach is outlined. Section 3.2 introduces the multi-criteria receiver-side relay election and presents a general analytic framework for performance evaluation of the relaying schemes. A demonstrative example of two-criteria based relay election priority is also presented here. Analytic and simulation results on two-criteria based relay election are contained in Section 3.3. Concluding remarks are drawn in Section 3.4.

3.1 Receiver-Side Relay Election Framework

Receiver-side relay election is a decentralized process where the next relaying node is decided through contention among all potential candidates [10, 12, 36]. Similar to 802.11 distributed coordination function (DCF), a variant of RTS/CTS (request-to-send/clear-to-send) message exchange is done between the transmitter and a potential forwarder before the data packet forwarding. However, unlike in 802.11, here the RTS packet is a broadcast massage containing position information of the sender and the final destination. Upon receiving this RTS packet, the potential relay candidates initiate a contention resolution process among themselves to elect the most suitable candidate as the next hop relay. The contention is typically resolved by introducing random or distance dependent time back-off. The first candidate to reply is the winner of the election process, and all other candidates abort.

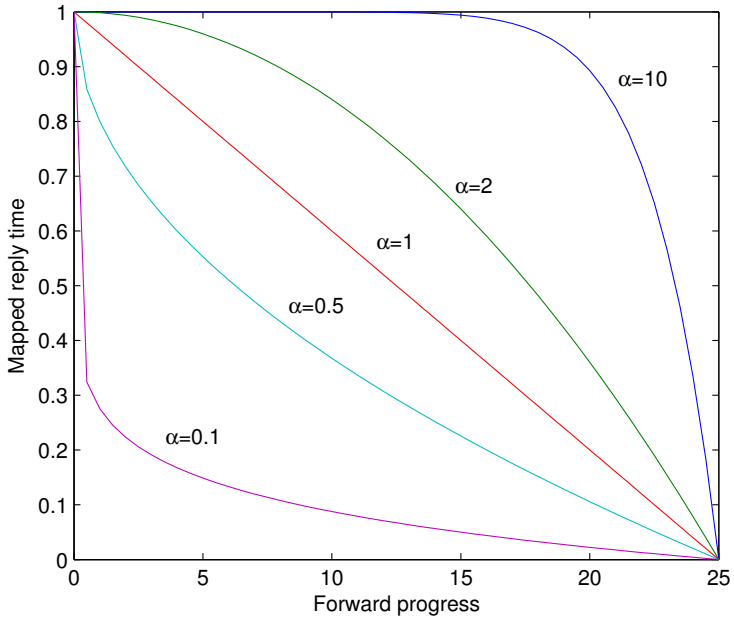


Figure 3.1 Parametric single-criterion mapping functions. The forward progress interval [0,25] is mapped onto the time interval [0,1]

Let us introduce a family of function $g_\alpha(.)$ that map, for each candidate i, the single election criterion, forward progress d_i, onto the response time delay X_i.

$$X_i = g_\alpha(d_i) = a(\alpha){d_i}^\alpha + b(\alpha) \tag{3.1}$$

where α is a shape parameter used to tune the performance of the election process. Here, perfect reception within a range $[0, R]$ is assumed, and the mapped time delay range is $[t_2, t_1]$. Equation (3.1) is obtained by generalization of the linear mapping function (see Figure 3.1). Coefficients $a(\alpha)$ and $b(\alpha)$ are obtained using the limiting conditions for the worst candidate ($g_\alpha(0) = t_1$) and the best candidate ($g_\alpha(R) = t_1$),

$$a(\alpha) = \tfrac{t_2 - t_1}{R^\alpha}; \quad b(\alpha) = t_1 \tag{3.2}$$

3.1.1 Election Delay

An important performance characteristic of a receiver-side relay election is the time duration of each election round. The average time until reception of a CTS response at the transmitter depends on the probability distribution of the mapped value X_i, which in turn is a function of the shape parameter α.

The cumulative distribution function F_x and density function f_x of the individual scheduled time X_i's are derived from the chosen the decision criterion (in this case, one-hop progress). Let $Y = \min_i \{X_i\}$ be the random variable denoting the time when the transmitter receives a CTS, in case the election process is successful. The distribution of Y

is given by

$$F_Y(y) = \Pr[Y \leq y] = \frac{1-e^{-nF_X(y)}}{1-e^{-n}}$$
$$f_Y(y) = \frac{nf_X(y)e^{-nF_X(y)}}{1-e^{-n}} \tag{3.3}$$

where n is the average number of active forward direction neighbors. From the above distributions the average delay of a contention process $E[Y]$ is computed.

3.1.2 Failure probability

Another characteristic of receiver-side election is the likelihood of collision between contending potential relays. Collisions are possible among two or more candidates if their respective back-off times are very close. More specifically, there could be collision and possible failure of the election process if candidates i and j schedule respective response time X_i and X_j such that $|X_i - X_j| \leq \beta$, where the collision vulnerability window β depends on the physical characteristics of the radio transceiver (e.g., transmit to receive switch-over time). The probability of collision can be expressed as [10]:

$$P_f = 1 - (h \copyright S_Y)(\beta) \tag{3.4}$$

where © represents the correlation integral function defined by

$$(h \copyright S_Y)(t) = \int_{-\infty}^{\infty} h(x) S_Y(t+x) dx$$

$S_Y(y) = 1 - F_Y(y)$ is the survival function of Y, and $h(y) = \frac{f_Y(y)}{S_Y(y)}$ is the corresponding failure rate.

Although the average duration of the election process $E[Y]$ can be made arbitrarily small with small α, this also increases the probability of collision P_f during the election process. Considering the effective delay of a successful election process, an optimal shape that minimizes the duration of election rounds while mitigating the probability of collision can be found. The value of the optimal shape parameter α depends on the recovery or retransmission policy used in case of collision during the election process. If the election rounds can be represented by unlimited Bernoulli trials until successful relay election, the optimal α value can be obtained by minimizing the effective delay D_e (see [10] for details), which is given by

$$D_e = \frac{P_f}{1 - P_f} t_1 + E[Y] \tag{3.5}$$

3.2 Multi-criteria Based Relay Election

3.2.1 Optimality Notion in the Multi-criteria Case

As noted earlier, multi-hop forwarding based on the one-hop progress criterion can hardly be optimal because of the unreliable nature of wireless links and other nodal limitations, such as energy, buffer capacity, etc. However, as more than one decision parameters are considered, the ranking of an alternative candidate becomes less obvious than in the single criterion case. Consider for example Figure 3.2 where two criteria are used to select the best relay node. With respect to a particular node (node A), the relationship with any other candidate can be classified as follows:

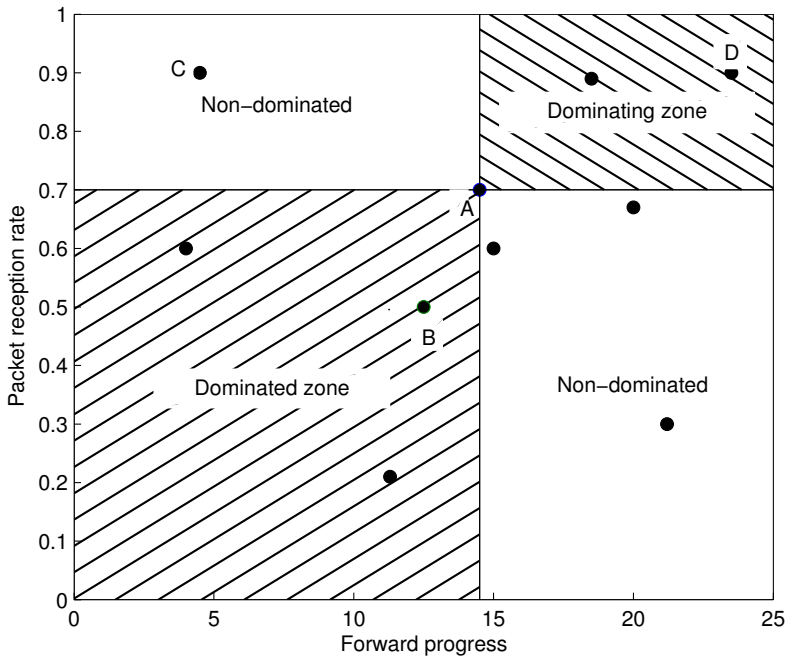


Figure 3.2 Relation between a particular node (A) and other candidates. In general, only candidates in the hatched areas can be strictly compared with node A.

- All nodes in the *dominated zone* are clearly strictly 'inferior' compared to node A because they perform strictly poorer on at least one criterion and at most as good on all others.

- All nodes in the *dominating zone* are clearly strictly 'superior' compared to node A because they perform strictly better on at least one criterion and at least as good on all others.

- However, nodes in the two *non-dominated* zones perform better than node A on a single criterion and poorer on all others. Therefore nodes in the non-dominated zone cannot be qualified as 'inferior' or 'superior' to node A.

Note that a forwarding decision can be made that maximizes all decision criteria whenever there exists a single candidate that dominates all others candidates (see node D

in Figure 3.2). However, in general, a single dominating candidate does not always exist and additional model is needed to define preference and tradeoffs among multiple criteria.

3.2.2 Multi-criteria Mapping Function

Similar to the single criteria case, a general preference model is introduced in the form of an aggregating function that combines all criteria into a single metric used to rank relay candidates. Ranking relay candidates based on multiple criBecause the order induced by the dominance relationship on the set of alternative candidates is partial, there may exist among the set of alternatives, pairs of mutually non-comparable candidates. With the mapping function, the objective is to introduce a single ranking scale through the use of an aggregating function that weights all criteria into a single one. Consider a decision based on k numerical criteria for which each candidate i has a performance index represented by the vector $\bar{\Omega}_i = (\Omega_{i1}, \Omega_{i2}, \cdots, \Omega_{ik})$. Without loss of generality, it is assumed that decision criterion (Ω_i) has a value in the range $[0, {\Omega_i}^{\max}]$ and has to be maximized.

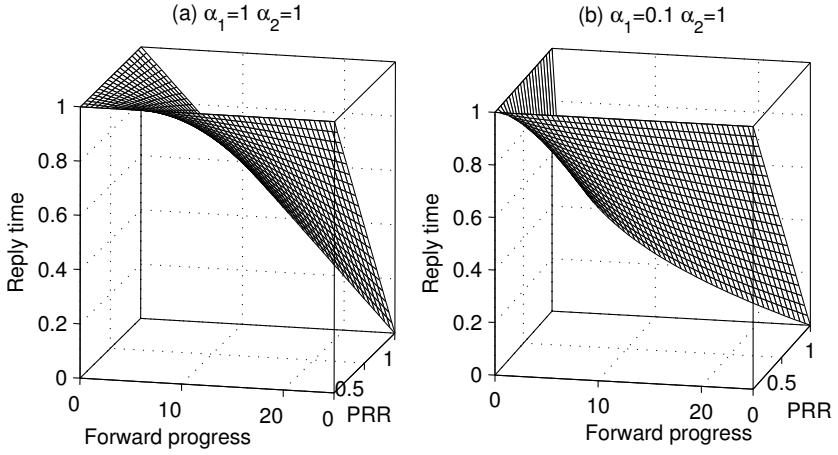


Figure 3.3 Mapping function in a two criteria case.

All decision variables are then mapped onto the scheduled time by introducing the multidimensional family of function (see Figure 3.3 for two criteria example).

$$g_{\bar{\alpha}}\left(\Omega_{i1}, \Omega_{i2}, \cdots, \Omega_{ik}\right) = a\left(\bar{\alpha}\right)\Omega_{i1}^{\alpha_1}\Omega_{i2}^{\alpha_2}\cdots\Omega_{ik}^{\alpha_k} + b\left(\bar{\alpha}\right) \tag{3.6}$$

where $\bar{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_k)$ is a k-parameter vector used to weight the k decision criteria. As in the single criterion case, the scheduled reply time for each candidate is $X_i = g_{\bar{\alpha}}\left(\bar{\Omega}_i\right)$.

From the perspective of transmitter-side relay selection, a corresponding cost metric can be derived from $g_{\bar{\alpha}}(.)$ as

$$C_{\bar{\alpha}}(\bar{\Omega}_i) = \Omega_{i1}^{\alpha_1}\Omega_{i2}^{\alpha_2}\cdots\Omega_{ik}^{\alpha_k} \tag{3.7}$$

Ranking all candidate with respect to $g_{\bar{\alpha}}$ (in descending order) or $C_{\bar{\alpha}}$ (in ascending order), creates a total ordering system on the set of all alternative candidates. That is, for any arbitrary two candidates i and j, $C_{\bar{\alpha}}(\bar{\Omega}_i) \leq C_{\bar{\alpha}}(\bar{\Omega}_j)$ or $C_{\bar{\alpha}}(\bar{\Omega}_i) \geq C_{\bar{\alpha}}(\bar{\Omega}_j)$.

Note that, for any positive real constant $m > 0$, $C_{m\bar{\alpha}}, m\bar{\alpha} = (m\alpha_1, m\alpha_2, \cdots, m\alpha_k)$ produces the same ranking as $C_{\bar{\alpha}}$. Therefore $g_{\bar{\alpha}}$ can be seen as a single *virtual criterion* ($C_{\frac{1}{\alpha_1}\bar{\alpha}}$) which, as in single criterion case in Section 3.1, is mapped onto the time interval $[t_2, t_1]$ for the purpose of receiver-side contention resolution:

$$g_{\bar{\alpha}}(\bar{\Omega}) = a(\bar{\alpha})\left[C_{\frac{1}{\alpha_1}\bar{\alpha}}(\bar{\Omega})\right]^{\alpha_1} + b(\bar{\alpha}) \tag{3.8}$$

Again, the parameter dependent coefficients are obtained from the limiting conditions for the worst and best candidates

$$a(\bar{\alpha}) = \frac{t_2 - t_1}{\Pi_1^k \left[\Omega_i^{\max}\right]^{\alpha_i}}; \quad b(\bar{\alpha}) = t_1 \tag{3.9}$$

As in the single criterion case, the multidimensional mapping function $g_{\bar{\alpha}}$ is a decreasing function with respect to each dimension considered individually.

3.2.3 Trading off Greediness for Link Quality

With the general mapping function presented in the above, the multi-criteria mapping is applied to an example case of forwarding scheme that finds an optimal tradeoff between link quality and greedy forward progress. An investigative approach is required because there is no a priori suggestion on what should be the optimal weights of the two criteria. For example, $\alpha_1 = \alpha_2 = 1$ gives $C_{(1,1)} = d_x * p_x$ (the product of one-hop progress offered by node x and the corresponding packet success probability), which corresponds to the normalized advance (NADV [34]) and maximum expected progress (MEP [35]). However, as will be presented in Section 3.3, the results show that this is suboptimal, and a substantially better network performance can be obtained by choosing appropriately the weighting parameters.

To see the impact of weight parameters (α_i) on the ranking of alternative relay candidates, consider node A (with $d_A = 14.5$ meter, $p_A = 0.7$) in Figs. 3.4. Note, how in case of $\alpha_1 = \alpha_2 = 1$ (Figs. 3.4(a)) a small increase in forward progress can compensate for a large decrease in link quality. On the other hand, with $\alpha_1 = 0.1$ and $\alpha_2 = 1$

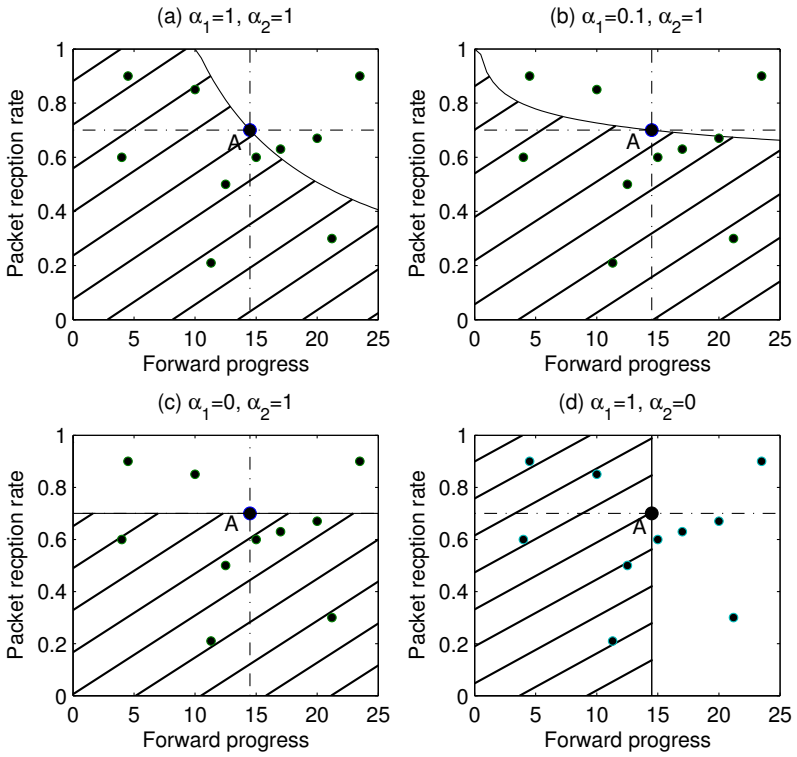


Figure 3.4 Preference relation with respect to a particular node. The set of alternatives is partitioned according to weight given to each criterion.

(Figs. 3.4(b)), a node at almost 10 unit distance away from A could offer an almost equally good alternative relay.

Note also that, to find the rules for optimal forwarding decision making, only relative values of the two weight parameters are needed. In other words, only the ratio $\frac{\alpha_1}{\alpha_2} \triangleq \lambda$ is considered for optimizing network performance (e.g., energy, packet failure, and delay) both from the perspective of transmitter-side relay selection and receiver-side relay election. The optimum value of λ is investigated via network simulations.

3.3 Simulation and Results

3.3.1 Simulation Model

Randomly deployed nodes are considered, with varied average density ρ (nodes/m^2). Nodal parameters have been based on Chipcon RFIC CC2420 operating with BFSK modulation scheme at 900 MHz. All nodes transmit with a nominal power (0 dB) and at a rate of 19.2 kbps. Log-normal fading channel with standard deviation of channel disturbance 4 dB and path loss exponent 4.0 have been assumed. Fixed path loss has been calculated considering near field distance 1 meter. Network performance has been studied with approximate end-to-end distance 100 meters. The scheduled reply times range from $t_2 = 250$ μsec to $t_1 = 1$ sec. Fixed packet size has been considered for all transmissions (50 Bytes for DATA and 4 Bytes for RTS). Each message is considered to have 100 data packets. No *a priori* transmission range has been assumed, all nodes capable of correctly receiving the initial broadcast RTS packet participate in the election process. Also, it has been assumed that a node is aware of its own geographic or virtual (hop-count based [46]) location information and that of the final destination. Each RTS packet contains position information of both the sender and the final destination.

3.3.2 Performance Metrics

End-to-end Packet Failure Rate To measure the relaying performance with a given tradeoff parameter through an unreliable wireless medium, packet failure rate is considered along the route. As a baseline comparison, the number of transmissions required for successful delivery of a message at the final destination are recorded. Fig. 3.5 shows the packet loss rate with node density, which indicates that beyond certain high node density, irrespective of the tradeoff parameter, the loss performance stabilizes. This is because

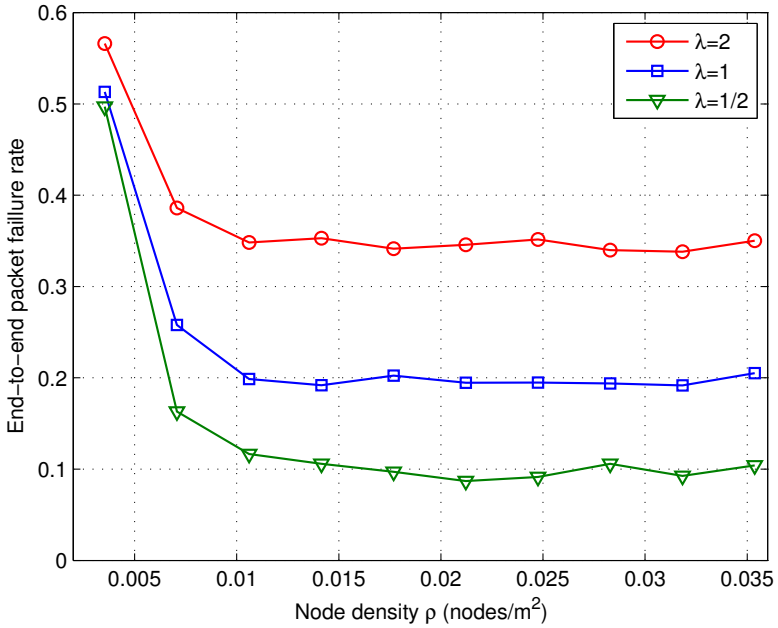


Figure 3.5 End-to-end packet failure rate as function of density.

at very low node density a node tends to find a relay that is associated with a highly error-prone channel. As the density increases, an optimum tradeoff is possible.

Fig. 3.6 shows that packet loss along the entire path can be reduced linearly with the tradeoff parameter λ. For example, $\lambda = \frac{1}{2}$ reduces the packet failure rate by 50% with respect to simple product of hop progress and packet success rate (i.e., with $\lambda = 1$).

End-to-end Forwarding Delay End-to-end delay due to packet transmission/retransmission are considered. In the simulation, once a relay node is elected, up to `max_retx` retransmissions are allowed. More than `max_retx` packet failures result in link error and a new relay election process is initiated. Also each successful transmission takes t_{tx} amount of time and each retransmission causes an additional delay t_{out} due to timeout (negative acknowledgment). Fig. 3.7 presents end-to-end packet delay as function of node density which shows the effect of packet

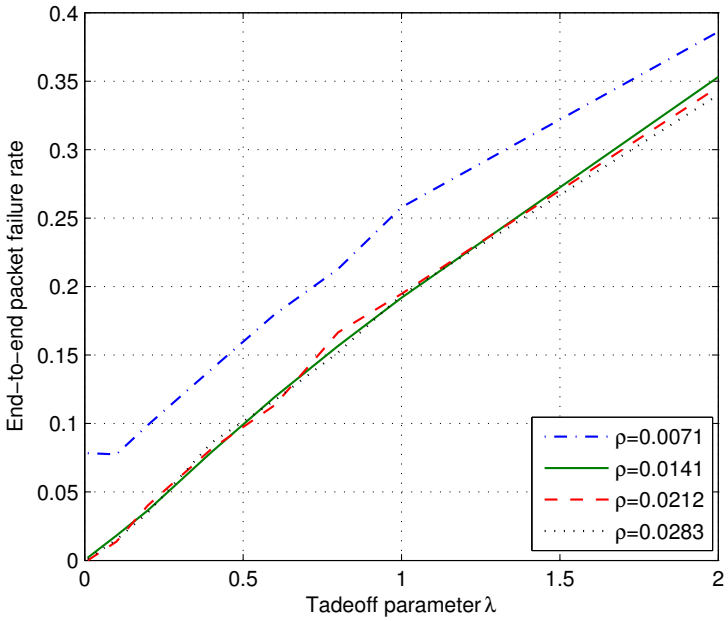


Figure 3.6 End-to-end packet failure rate as function of the tradeoff parameter λ.

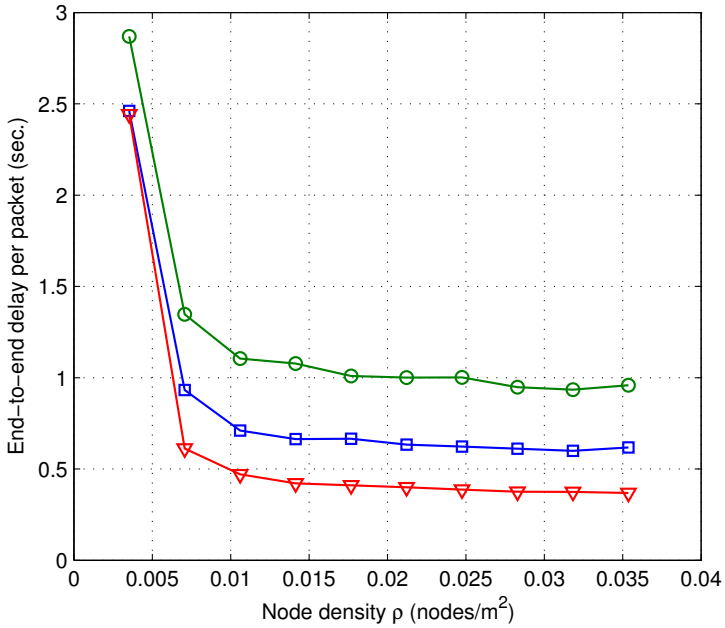


Figure 3.7 End-to-end delay as function of node density. `max_retx` $= 8$, $t_{tx} = 21.1$ msec, $t_{out} = 84.4$ msec.

failure on packet delay (compare Figure 3.7 with Figure 3.5). Although Figure 3.6 suggests that packet failure and, as a consequence, end-to-end delay can be made arbitrarily small

by selecting smaller tradeoff parameter λ, the next result on energy efficiency shows that there exists a minimum value of λ beyond which adverse energy effect can be seen.

End-to-end Energy Consumption The energy efficiency of a given forwarding strategy is evaluated by the number of transmissions required along the route for a successful end-to-end packet delivery. As expected, the energy requirement due to forwarding decreases with higher node densities, where it is more likely to find a neighbor offering a good combination of hop progress and link quality (see Figure 3.8). Fig. 3.8 also shows that it is possible to

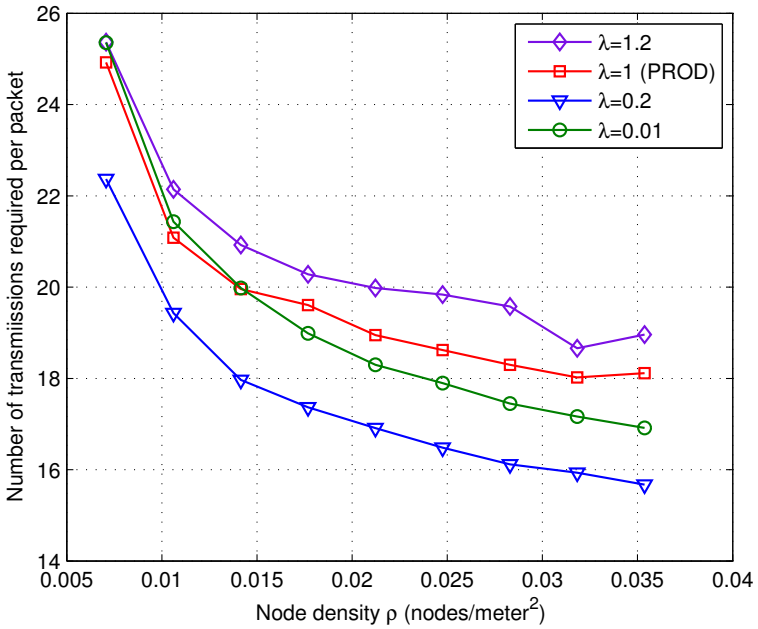


Figure 3.8 Energy consumption (number of required transmission) for end-to-end packet delivery as a function of node density

improve energy efficiency by reducing the weight given to hop progress. Clearly $\lambda = 0.2$ outperforms the simple product form ($\lambda = 1$). It can also be seen that further reduction of the weight given to hop progress result in increasing energy consumption. Fig. 3.9 depicts that an optimal tradeoff between hop progress and link quality can be found that minimizes the required energy consumption. It shows that the optimal performance is

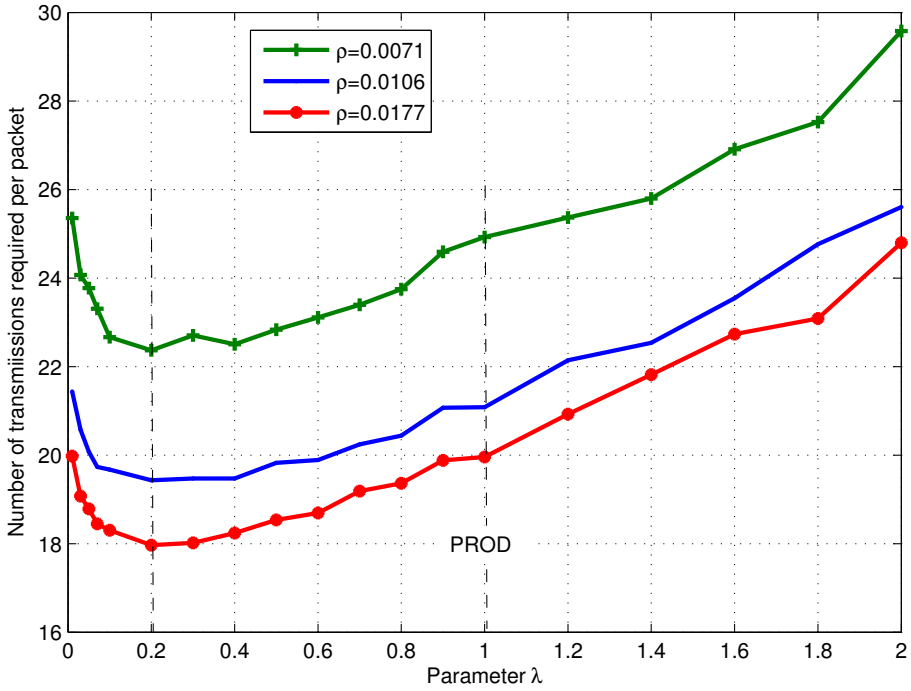


Figure 3.9 Energy consumption (number of required transmission) as a function of the weightage parameter α_1 (with $\alpha_2 = 1$)

achieved approximately at $\lambda = 0.2$. Notice from Figure 3.6 that this optimal λ can achieve up to *approximately 5 times reduction* in packet failure rate with respect to the simple product form (i.e., with $\lambda = 1$).

3.4 Conclusions

This chapter presented multi-criteria receiver-side relay election framework for multi-hop relaying in ad hoc networks. Via intuitive reasoning and examples the importance of finding optimum weighted relay election/selection criteria has been qualitatively shown. A generalized cost metric in the form of multi-parameter mapping function has been proposed and used to investigate optimal tradeoff between greedy forwarding and link quality. It has been shown that a much better network performance in terms of total energy consumption

for successful end-to-end routing can be achieved via judicious selection of the weighting parameter that optimally trades off between greediness and link quality. The multi-criteria mapping function is quite general and can also be applicable to transmitter-side relay selection process.

CHAPTER 4

MESH BASED FORWARDING

Reliable mesh communications in dense wireless ad hoc networks require the creation of both self organizing mesh structures and mesh routing protocols to accomplish efficient and reliable communications with the added infrastructure redundancy. To date, much of the research in the area has focused on communication protocol design. The investigations often are based on a mesh network structure already fully formed and some times fixed to the underlying physical node topology. Therefore, there is a need for a platform to build mesh networks with structural flexibility and to provide management functions to network- and application-level protocols. This paper presents the distributed diffusion-based mesh (DDM) algorithm for distributed mesh construction that instructs distributed nodes how to make the desired connections with their neighbors. This is accomplished by introducing the concept of *connection rule*, which defines allowed connections at each mesh node, combined with a *token* signal that initiates and controls the structure and boundaries of the resulting mesh. It is argued that slight changes in mesh network structure greatly affect network performance and show how the combined use of rule and token signal offers control over the resulting mesh structure. This methodology can be used for cross-layer optimization to achieve a network topology suitable for different network applications. Compared with aspects of existing protocols, the proposed algorithm also provides a large reduction in communication overhead.

4.1 Introduction

In recent years, wireless mesh networks have received an increased amount of interest from academia and industry. Many communities and private companies are field-testing and deploying IEEE 802.11-based wide area mesh networks [47–49]. The vast majority of the reported networks are considered pre-built and deployed with limited self-organization.

It is not well known how complex it is to accomplish self-organization in a large wireless mesh networks. In the current literature, two implicit assumptions are often made with respect to the mesh formation phase: (i) either the mesh is small, and it is assumed that it can be constructed in a centralized manner before the network is brought to life, or (ii) the mesh is large and it is formed by allowing all nodes to make all possible connections available. However, in many applications where nodes can be arbitrarily spread on an area of interest, a fully interconnected network may not be realistic nor efficient. As a result, these meshes are either small and cannot be applied to large scale situations or their structure is fixed by the underlying physical deployment of nodes. It is therefore desirable not to limit the structure of the resulting mesh network to the physical node deployment. For example, in multi-channel networks, channel assignment strategies may require that a node connects with a specific subset of its neighbors to mitigate interference and improve throughput. Furthermore, the capacity of wireless networks is limited by the number of active wireless nodes and the channel use strategies [50–55]. The influence of metrics such as nodal degree, links, and nodes density, mesh performance has been widely researched but the impact of mesh structure over network performance and reliability has been barely reported.

In this chapter, a generic diffusion-based mesh algorithm is proposed, which provides dynamic mesh construction and management services to routing and upper

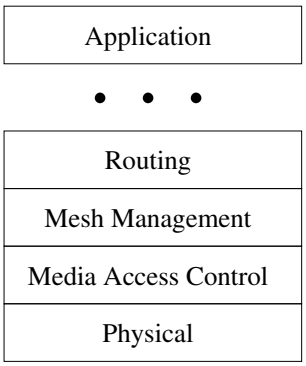


Figure 4.1 Place of the Mesh Network Layer in the (OSI) network model.

layer protocols (see Figure 4.1). Distributed diffusion-based mesh (DDM) methodology is a distributed protocol that allows dynamic configuration of mesh structure through instructions diffusion. These instructions define which links are allowed between neighboring nodes and the extension (boundaries) of the mesh network.

Simulation results of the proposed DDM algorithm shows a large reduction in communication overhead when compared with existing routing schemes, which are a currently considered alternative with slightly similar objectives. It is shown that with comparable nodes and links density, as well as equal average nodal degree, different mesh structures produce vastly differing network performance and reliability results.

4.2 Motivation

It is widely understood how mesh network metrics, such as nodal degree, hop count, and number of nodes, impact mesh network performance. However the impact of mesh structure on network performance and reliability is less clear. For example, it is well known that network throughput increases as nodal degree increases. However, it is not always possible (or desirable) to increase nodal degree because of other constraints, such

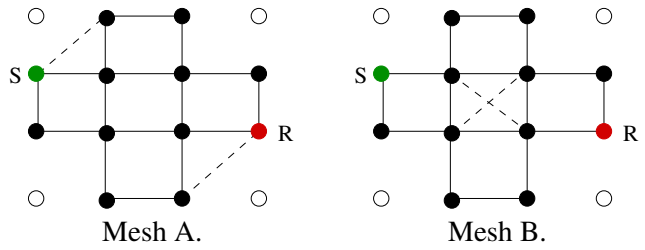


Figure 4.2 Two similar mesh networks with structural differences.

as available channels and multi-user media access. In such cases, performance can be improved through careful selection of the mesh structure.

For example, Figure 4.2 presents two meshes that are similar in all regards (average nodal degree, number of nodes, number of links, etc.) except for their structures. Even though all other characteristics are equal, these two meshes perform differently because of their structures, as shown by Figure 4.3. Without any calculation, It can be seen that there are much more distinct paths from S to R in the mesh B (on the right) than in mesh A (on the left). Next, consider a simple multi-path routing scheme which sends a copy of the packet to transmit simultaneously over all outgoing links. Assume that this routing policy is used with forward error correction (FEC). This means that no packet retransmission is used at the link layer level. Instead, any error is detected at the final destination and possibly corrected. Figure 4.3 shows the throughput performance of this simple routing protocol running over these meshes. An improvement of up to 35% in throughput is observed between mesh A and mesh B. Thus, the structure of a mesh significantly affects the performance of network protocols.

4.3 Distributed Diffusion Mesh Construction Algorithm

To establish the basic concepts of the proposed distributed diffusion mesh building algorithm, the physical node deployment is assumed to be regular and rectangular. Consider

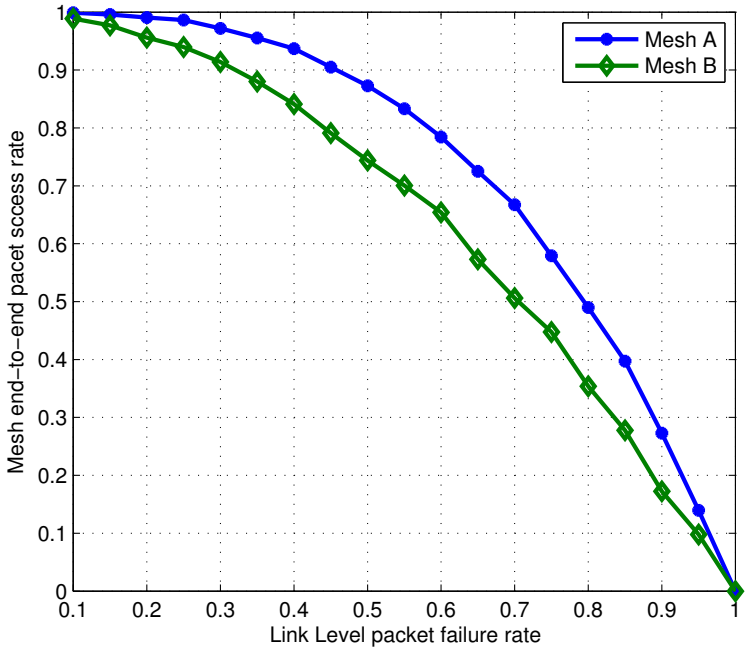


Figure 4.3 Comparison of end-to-end packet success rates of mesh A and mesh B.

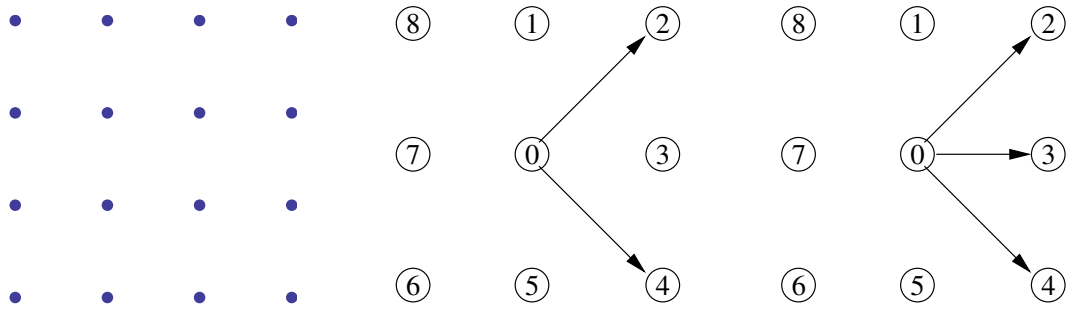
the physical node layout in Figure 4.4(a) on which the mesh network is to be overlayed. Each participating node may establish communication links to all or a subset of its neighbors. In this setting, a mesh can be fully defined (and constructed) in the following manner:

(i) A starting point;

(ii) A rule for establishing connections (applied recursively at every node);

(iii) A boundary condition to determine when and where to stop.

The main objective of DDM is to accomplish these steps in an efficient and distributed fashion. The construction of the mesh is carried out as command injected at the starting point - step (i) - and diffused throughout the network. The propagated command carries the connection rule - step (ii) - and a token signal - step (iii). Below, the role and



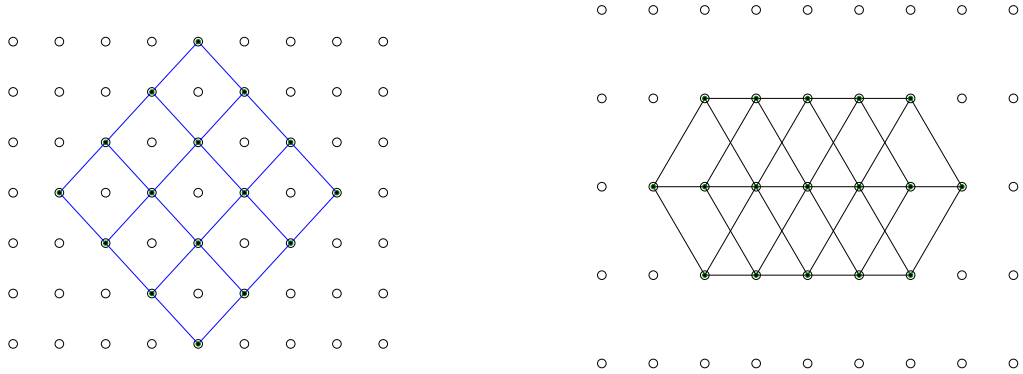
(a) Underlying grid (b) Rule 10 (c) Rule 14

Figure 4.4 Representation of connection rules in DDM as a string of 8 bits: (a) Underlying grid, (b) example of rule 01010000, and (c) example of rule 01110000.

implementation of the connection rule and token signal are first described followed by an illustration of their combined use in the DDM algorithm.

4.3.1 The Connection Rule

As specified by step (ii), the connection rule of the mesh construction algorithm indicates connections that are permitted at each mesh node. Consider again wireless nodes distributed in a rectangular regular grid (Fig. 4.4(a)). From the perspective of each mesh node, there are 8 available neighbors to which a connection can be made (Fig. 4.4 (b and c)). Each neighbor is assigned a bit which is switched (ON or OFF) depending on whether a connection to this neighbor is allowed or prohibited. Similar to the methodology used in [56], the rule is represented by an 8-bit string that indicates whether a connection for the corresponding neighbors is allowed. For example rule 10, in Figure 4.4(b), which has the binary form 01010000, allows connection only with upper right and lower right corner neighbors. Similarly, rule 14, with binary 01110000, allows connections to all neighbors on the right.



(a) Sample mesh by rule 10 (b) Sample mesh by rule 14

Figure 4.5 Sample mesh networks constructed with (a) rule 10 and (b) rule 14.

A total of 256 distinct types of elementary regular rectangular-grid based meshes can be identified. These types can be combined to form more complex mesh structures. Figures 4.5(a) and 4.5(b) show meshes formed by application of rule 10 and rule 14, respectively.

4.3.2 The Token Signal

The token signal primarily determines the boundaries of the resulting mesh by imposing limits on the how far the mesh contraction command travels in each direction. The signal is made of a series of time-to-live (TTL) counters representing the maximum progress of the token in each direction starting from the current position. For a 2-dimensional rectangular grid based mesh, four counters are used, one for each of the four main directions

$$T = (TTL_{up}, TTL_{down}, TTL_{right}, TTL_{left}.) \tag{4.1}$$

The four remaining diagonal directions are expressed as combination of the main directions. For example, while a move in the *up* direction decreases only TTL_{up}, a move in the *uperright* direction decreases both the TTL_{up} counter and the TTL_{right} counter.

More formally, the four main directions are the bases of a 4-dimensional vector space,

$$\begin{aligned} D_{up} &= [1\,0\,0\,0]\,, \\ D_{right} &= [0\,1\,0\,0]\,, \\ D_{down} &= [0\,0\,1\,0]\,, \\ D_{left} &= [0\,0\,0\,1]\,. \end{aligned} \tag{4.2}$$

The diagonal directions are defined as

$$\begin{aligned} D_{upperright} &= [1\,1\,0\,0]\,, \\ D_{lowerright} &= [0\,1\,1\,0]\,, \\ D_{loweleft} &= [0\,0\,1\,1]\,, \\ D_{upperleft} &= [1\,0\,0\,1]\,. \end{aligned} \tag{4.3}$$

4.3.3 Algorithm Operation

Figure 4.6 presents the flow of execution of DDM from the perspective of a single node. The execution of the algorithm is initiated by injection of the initial token signal at the starting point of the mesh. Figure 4.7 shows an example where the initial token $[2420]$ is

injected at node 1. Rule 10 (binary 01010000) is used for this example. In other words, only connections upper-right and lower-right neighbors are allowed.

Upon the reception of a token signal, a node checks the TTL's contained in the token signal to determine if their current values permit further propagation of the token in the directions allowed by the rule. If this is the case, the node makes the allowed connections and sends updated token signal to the corresponding neighbors. In this example, node 1 connects with nodes 2 and 3 and sends them appropriately updated token signals.

The token is updated for each direction by subtracting the vector corresponding to the direction (as defined in equations 4.2 and 4.3) from the token vector. For the upper-right neighbor (node 2), node 1 sends the updated token $[2\ 4\ 2\ 0] - [1\ 1\ 0\ 0] = [1\ 3\ 2\ 0]$. Similarly, it sends $[2\ 4\ 2\ 0] - [0\ 1\ 1\ 0] = [2\ 3\ 1\ 0]$ to its lower-right neighbor (node 3).

Upon reception of the updated token, nodes 2 and 3 carry out the same process, connect to their own selected neighbors and propagate further a yet updated taken signal. The process continue until all the TTL's in the token are reduced to zero or their values do not allow any further progression in the direction allowed by the rule (see node 9 in figure 4.7).

4.4 Evaluation

The objective of DDM is to offer mesh construction and management services to other network or application layer protocols. An approach to solve a comparable problem is the *ad-hoc on demand distant vector with mesh multi-path* (AODV-MM) protocol [45]. It should be noted that AODV-MM is a complete mesh routing solution based on the widely used ad hoc on demand distant vector (AODV) protocol. The comparison is limited to the mesh building phase of AODV-MM.

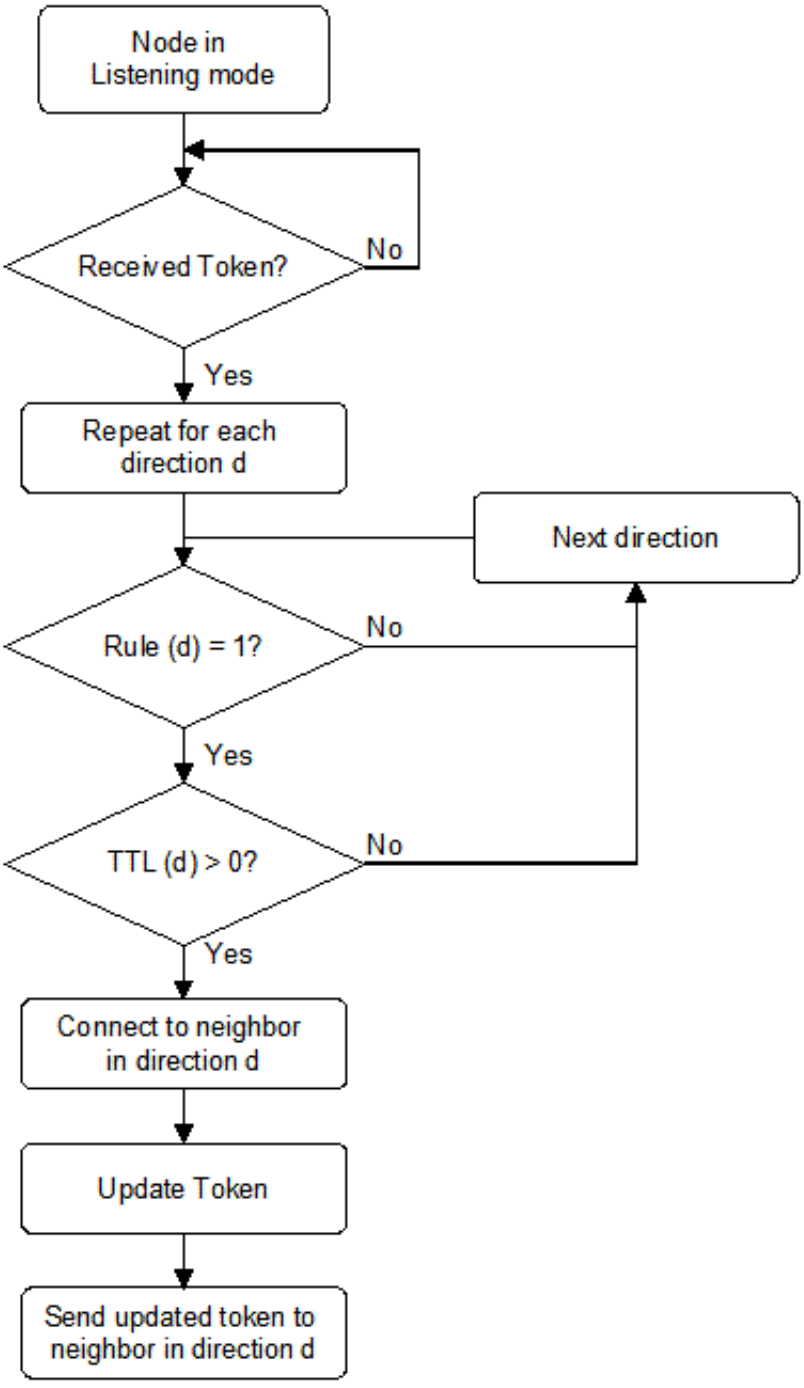


Figure 4.6 Flow of decision for the establishment of link at each mesh node.

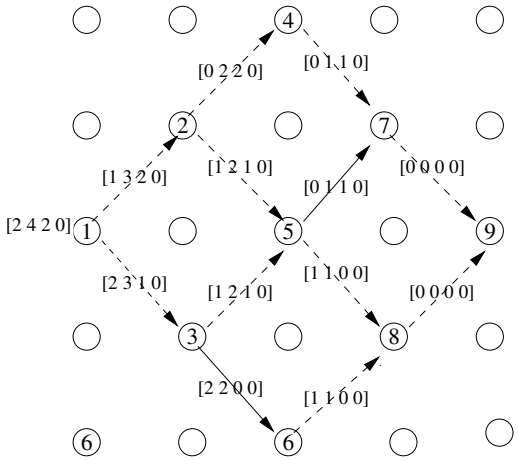


Figure 4.7 An example token diffusion by DDM. Rule 10 is used with initial token [2 4 2 0] injected at node 1.

Table 4.1 Simulation Parameters

Parameter	Value
Total area	800m x 800m
Node transmission range	25m
Medium access scheme	CSMA with collision avoidance

4.4.1 Simulation Settings

This section describe the settings used in the simulations. First, a brief overview of AODV-MM. In [45], the authors of AODV-MM used the mesh type shown in Figure 4.5(b) to demonstrate the merits of their proposed mesh multipath routing protocol. By using a combination of route discovery request signals and overhearing, each node is able to maintain a primary route and a series of secondary routes to the destination. In DDM the same mesh is constructed by using rule 10 and an initial token vector of $[1\ n\ 1\ 0]$, where n represents the number of hops separating the source S from the destination D. All comparisons with AODV-MM were made using this mesh.

The simulation network is akin to a sensor network environment with rectangular regular grid deployment over an area of 800 m $\times$800 m. All network nodes are considered homogeneous with the same resources and transmission range. Table 4.1 lists additional simulation parameters used.

4.4.2 Performance Metrics

In dynamic and resource-constrained network environments, the main requirement for communication protocols is to minimized control overhead and reduce communication delay. Three performance metrics are considered for the evaluation: (1) communication overhead, (2) time delay, and (3) network disturbance.

- ***Communication Overhead***: Communication overhead is defined as the amount of communication required for the construction of a specific network-wide mesh. In the simulations, this is captured as the total number of transmissions (or broadcasts) required for the establishment of the mesh network.

- ***Time Delay***: This represents the time added by the execution of the algorithm to other network operations. Special attention is paid to two variants of the delay. The first captures the time at which the controls signals are injected at the starting point reach the end point (destination) for the first time (to have an idea of the speed at which the protocols find a mesh multi-path to the destination). The second captures the total completion time as the time lapsed until all copies of the control packets and their modified versions dissipate from the network. This gives an idea of the level of disturbance and indirect network delays the protocol may cause. In the simulations, all time delays are captured as number of simulated timed slots.

- ***Network-wide footprint***: This metric represents the number of nodes which participated in the building of the mesh structure, whether or not they belong to the

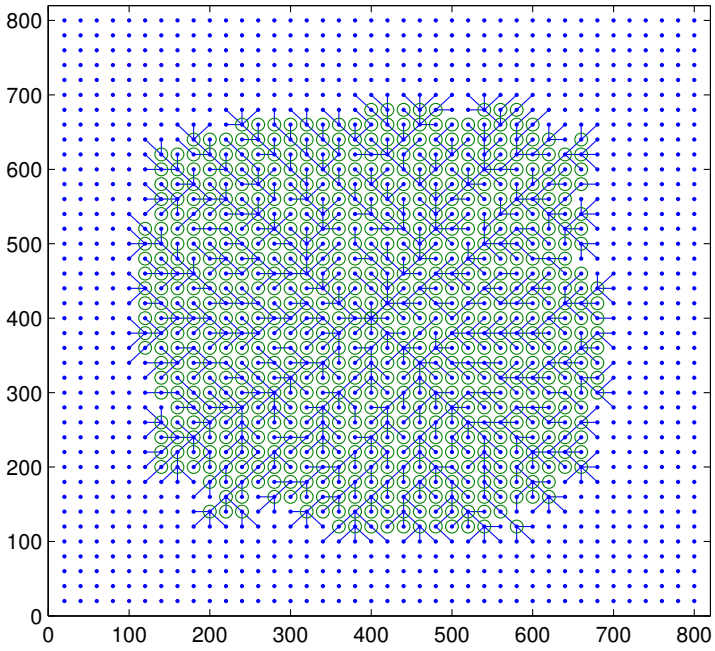


Figure 4.8 In AODV-MM, the diffusion of control signal is done in all directions which puts a burden on a large portion of the network.

final mesh. In the simulations, this is captured by counting the number of nodes that have either received or overhead mesh construction related communications.

4.4.3 Results and Discussion

Communication Overhead Figure 4.10 shows the communication overhead of mesh building under AODV-MM and DDM. The x-axis represents the length of the mesh (hop count of the shortest path between source and destination nodes). It is noted that DDM has a small overhead compared to that of OADV-MM for which the communication overhead grows exponentially with the length of the mesh. This is mainly the result of the omnidirectional property of the route request dissemination process used by AODV-MM, itself inherited from the original AODV protocol.

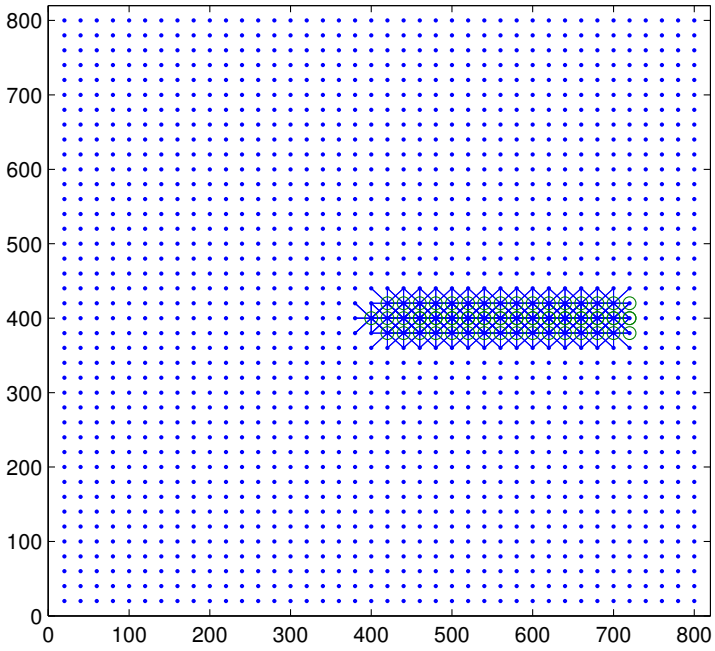


Figure 4.9 DDM keeps a minimal network-wide footprint.

This reduction in communication overhead is an important consequence of the use of direction-based connection rules and TTL tokens. Control signals which would have been propagated in all directions are focused only in the directions in which the mesh should be created. A drawback, however, with the use of directional diffusion is that knowledge of the approximate location of the destination is required. However, this is often the case in many ad hoc networks (especially sensor networks) where the location of important destinations such as access points or data sinks are known.

Time Delay Figure 4.11 shows that while both protocols complete the construction of the mesh in similar times, it takes longer for packets generated by AODV-MM to vacate the network. This is another manifestation of the use of directionally focused control messages.

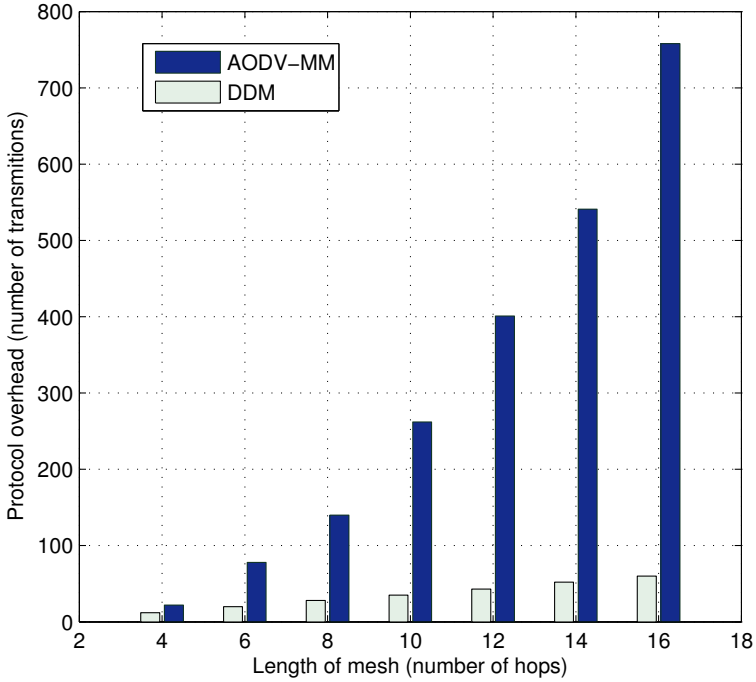


Figure 4.10 Communication overhead of AODV-MM and DDM.

Because DDM uses fewer control messages, it take a shorter time to complete transmission of all copies, thus liberating network time and resources for other use.

Network-wide Footprint Another important metric is the network-wide footprint. Figures 4.8 and 4.9 show active zones within the network when the protocols were used to create a mesh similar to figure 4.5(b) over 16hops. The performance gain can clearly be observed. Only a limited number of nodes are involved in the case of DDM.

4.5 Conclusions

A distributed diffusion-based mesh (DDM) algorithm is introduced to build meshes in a distributed manner. The objective of DDM is to offer mesh construction and management services to other network or application layer protocols. It is shown that with comparable

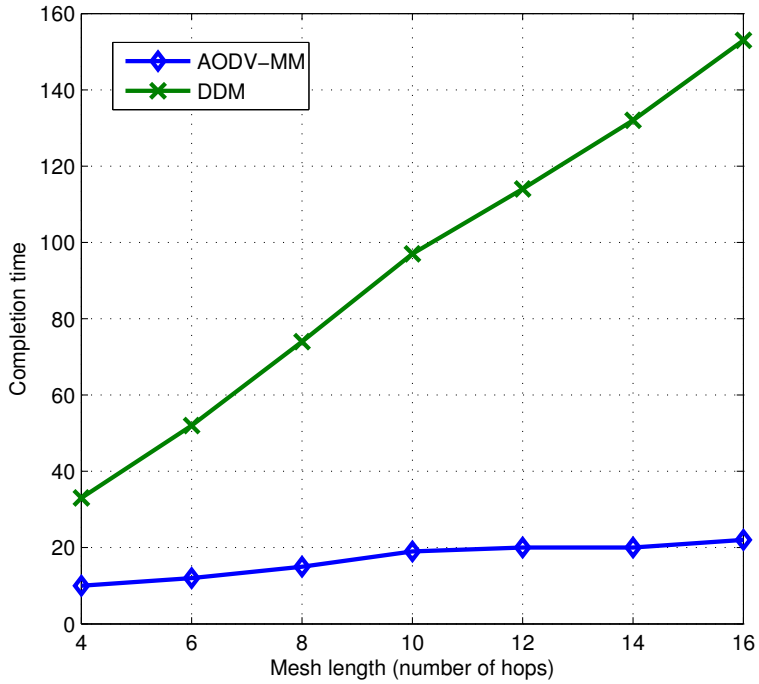


Figure 4.11 Maximum lifetime of control packets inside the network.

nodes and links density, as well as with equal average nodal degree, different mesh structures produce vastly different network performance and reliability results. Thus DDM's ability to produce a wide variety of mesh structures can serve as a tool to study these mesh structures and evaluate their performance under various network protocols.

CHAPTER 5

SYSTEM-LEVEL POWER CONSUMPTION ANALYSIS

5.1 Introduction

The design of communication protocols for battery-powered wireless devices faces many challenges that are unique to resource-constrained devices. Limits on channel bandwidth, memory resources, and processing capabilities play major roles, but a fundamental constraint that needs major attention is the limited energy resource. In many sensor network applications, the nodes cannot be replenished once their battery is drained. For other portable devices, there is an increasing demand for long battery life between recharges. An accurate model of power consumption is therefore needed to aid in the design of power-efficient protocols and applications.

Power efficiency is particularly critical for sensor devices and it has received a lot of attention by the sensor-network research community [55, 57–66]. A dynamic power management scheme proposed turning the operating system on and off intelligently to minimize energy wastage [67]. It has been observed that overhearing is a major source of energy consumption in sensor networks [68, 69]. Sensor medium access control (S-MAC) extends the concepts of *request to send* (RTS) and *clear to send* (CTS) in 802.11, and reduces the data transmission coordination-related energy consumption by locally synchronizing at the beginning of every activity cycle [68]. It has been demonstrated that Aloha-like access protocol with preamble sampling operates power efficiently by reducing idle listening energy cost [70, 71]. Based on the understandings of power consumption in various node states, a low-power medium access protocol named B-MAC has been recently

proposed [72]. B-MAC is based on a preamble sampling approach and provides the system designers with the flexibility of choosing between energy awareness and service criteria.

Analytical models of power consumption have received some attention, focused mainly on system performance and energy optimization [73, 74]. These works use the continuous-time Markov decision process (CTMDP) to characterize power consumption in different states and to search for the optimum control policy as a means to automate the node. Although CTMDP models are suitable for optimal control problems in which the primary goal is to find an optimal policy that can be implemented as a software or hardware component, their applicability is limited by the requirement of exponential distribution of the inter-decision time. Discrete-time Markov decision process (DTMDP) state model has been used to model sensor nodes as a sleep and active controlled dynamic system [75]. The authors showed that, by controlling the state of nodes, the optimal interference and routing performances can be achieved. In both CTMDP and DTMDP models, the dwell time in a state (also called sojourn time) is assumed to have a memoryless distribution. Optimal node placement strategy in a one-dimensional sensor network was studied in order to achieve energy-optimized system performance [76].

This chapter presents a model of the power consumption in which sensor nodes are modeled as dynamic systems influenced by a power management policy and a medium access protocol. The wireless node is model as a semi-Markov decision process (SMDP) with five states: *transmit* (*Tx*), *receive* (*Rx*), *MAC contention*, *idle*, and *sleep*. The sojourn times and state transitions are associated with a device-specific power consumption cost. Because the states and transitions are dependent on activity and channel condition, the overall operation of a communication protocol is characterized as the evaluation of a given policy of the SMDP. The energy efficiency of the protocol is measured as the long-run

average cost per unit time of the policy. It is shown that the developed model can be used as an optimization tool for a given communication protocol. Using manufacturer-provided data [77], the model is applied to the MICA2 sensor motes equipped with the Chipcon CC2420 radio transceiver (see Table 5.1). The optimum sleep time of nodes and data packet size for maximizing transmission success rate with a minimum energy cost is estimated. The model can also be used in comparing the energy efficiency of different communication protocols in sensor networks, and it can provide a guideline to determining the energy consumption pattern for a given node and sink deployment strategy.

The remainder of this chapter is presented as follows. Section 5.2 presents model assumptions, notation conventions, and backgrounds. Section 5.3 presents the construction of an SMDP-based power consumption model. Section 5.4 presents the application of the model to the MICA2 sensor nodes. Section 5.5 shows analytical and simulation results. Concluding remarks are drawn in Section 5.6.

5.2 Preliminaries

5.2.1 Dynamic Stochastic Systems

Dynamic stochastic systems are the systems that evolve with time under the influence of a control entity or random events. In general, a dynamic stochastic system can be classified as (i) continuous-time and (ii) discrete-time, if the system is observed in continuous and discrete-time, respectively. Another system classification is (i) controlled and (ii) uncontrolled, depending on the existence or absence of an identifiable system controller.

In all classifications, the system is observed at the initial time $t_0 = 0$ and is found in one of a finite (possibly countably infinite) number of states $X_0 = i,\ i = 0, 1, 2, \cdots$. After a sojourn time of τ_0, the system jumps into another state j ($X_1 = j$) at the time instant

$t_1 = t_0 + \tau_0$ with probability p_{ij}, stays for a time τ_1 before jumping to another state, k ($X_2 = k$), at time $t_2 = t_1 + \tau_1$ with probability p_{jk}, and so forth. The two processes $\{X_\ell\}$ and $\{\tau_\ell\}$, $\ell = 0, 1, 2, \cdots$, constitute the stochastic model of the dynamic system.

In a controlled dynamic system, a controller or a decision-maker is allowed to influence the system by choosing at each decision epoch one of a finite (possibly countably infinite) number of actions a_κ, $\kappa = 0, 1, 2, \cdots, K$ $(K \leq \infty)$ The evolution of the system among states and the corresponding sojourn times within states are therefore determined by the actions of the decision-maker. In other words, the distribution of the two processes $\{X_\ell\}$ and $\{\tau_\ell\}$, $\ell = 0, 1, 2, \cdots$, becomes action-dependent.

The theory of controlled stochastic systems offers practical analytical frameworks for the study of (i) optimal control problems and (ii) performance evaluation problems. While the former is the primary concern of DPM solutions [73, 78, 79] that seek to find an optimal decision strategy (or optimal policy) among all feasible decision strategies, the later focuses on the evaluation of a given decision strategy. In both cases, a cost (or reward) structure is associated with a strategy, wherein each state and state transition of the system incurs a different level of system cost. The primary focus of this chapter is the performance evaluation of a given strategy (communication protocol).

5.2.2 Semi-Markov Decision Processes (SMDP)

Consider a controlled stochastic dynamic system as introduced earlier. The system with state space $\mathcal{I} = \{0, 1, 2, \cdots, I\}$ is controlled by a sequential decision-maker with action space $\mathcal{A} = \{a_0, a_1, a_2, \cdots, a_K\}$. The decision-maker reviews the state of the system at given (possibly random) epochs and takes a decision. In each state $i \in \mathcal{I}$, a set of actions

$A(i) \subset \mathcal{A}$ is allowed. As a consequence of selecting an action at a decision epoch, a reward (or cost) is incurred.

Denote the system state X_0 at time t_0, and let t_ℓ and $X_\ell,\ \ell = 0, 1, 2, \cdots$, be the subsequent decision epochs and the corresponding system states, respectively.

Definition 4 *The above-defined model is said to be an SMDP if the embedded process* $\{X_\ell\},\ \ell = 0, 1, 2, \cdots$, *has the Markov property, i.e., the state of the system at the next decision epoch depends on the history of the system only through the current state.*

In other words, at each decision epoch, the time until the next decision and the next state of the system only depends on the current state and decision currently chosen by the decision-maker. Note that there is no restriction on the distribution of the inter-decision time. Discrete-time Markov decision process (DTMDP) and continuous-time Markov decision process (CTMDP) are special cases of SMDP with fixed inter-decision time and exponentially distributed inter-decision time, respectively. Below, the additional definitions relating to SMDP and the notational conventions used in the chapter are presented.

Definition 5 *A* decision rule *of an SMDP specifies the rule for selection of actions in each state at a specified decision epoch.*

- *Deterministic decision rules are formally defined as function* $D_t : \mathcal{I} \rightarrow \mathcal{A}$ *which specifies that the action* $D_t(i)$ *is chosen with certainty if the system is found in state* i *at the decision epoch* t*, for all* i*.*

- *Randomized decision rules prescribe that, when the system is found in state i, action $a_\kappa \in A(i)$ is chosen with probability $p_i(a_\kappa, t)$, where*

$$\sum_{a_\kappa \in A(i)} p_i(a_\kappa, t) = 1.$$

With stationary randomized policies, $p_i(a_\kappa, t)$ is simply $p_i(a_\kappa)$. Correspondingly, action-dependent state transition probability from state i to j is denoted as $p_i(a_\kappa, j)$.

Definition 6 *A* decision policy *specifies the decision rule of the decision model at all decision epochs. Denote $f = \{D_0, D_1, D_2, \cdots\}$ to be the decision policy that applies decision D_ℓ at the ℓ^{th} decision epoch t_ℓ.*

- *A decision policy is said to be deterministic (respectively, randomized) if it applies a deterministic (respectively, randomized) decision rule at all decision epochs.*

- *A decision policy is said to be stationary if it applies the same decision rule at all decision epochs $f = \{D, D, D, \cdots\}$.*

5.3 SMDP-Based Power Consumption Model

For the purpose of performance analysis, the model is sufficiently defined by (i) the state space, (ii) the action space along with the action selection policy in each state and the associated state transition rates; and (iii) the cost/reward structure associated to the states, actions, and state transitions.

5.3.1 State Identification

The model states are determined based on the system performance. For the power consumption, every distinct power consumption level is associated with a model state. For the case of sensor devices operating under ALOHA-like medium access control, five states are identified: *sleep*, *idle*, *wait*, *transmit*, and *receive*. Table 5.1 shows the current drawn by a sensor node in various communication and power-saving modes of the radio transceiver.

Table 5.1 Typical Power Consumption Data (input current drawn) of Chipcon CC2420.

Mode	Current drawn
Transmit:	
0 dBm (max)	17.4 mA
-25 dBm (min)	8.5 mA
Receive	19.7 mA
Idle (oscillator and voltage regulator on)	426 μA
Sleep (voltage regulator on)	20 μA

5.3.2 System Controller and Actions Model

The behavior of a dynamic system in a given state can be characterized by the distribution of the time spent in the state (dwell time) and the transition probabilities to subsequent states. Practical systems are seldom amenable to analysis where the state dwell time can be modeled with a single distribution, such as a simple exponential distribution. Instead, the behavior of the system in a state is best characterized as one of many possible alternatives. For example, when a wireless node is in transmit state, the amount of time it spends in this state, and the likely next state, depends on the type of packet (data or acknowledgement) currently being transmitted.

In controlled dynamic systems, the alternatives in each state are actions actively chosen by a system controller [73]. However, these alternative characterizations of the system behavior in a given state need not be based on active actions taken by a physical or

software-based controller. Instead, because of the flexible mathematical formulation of an SMDP (as described in Section 5.2), any specialization of the system behavior in a state is modeled as an action that influences the system's dwell time and transmission rate to other states.

5.3.3 Energy Cost Structure

Every pair of allowable state action/event pair $(i, a_\kappa)\ i \in \mathcal{I},\ a_\kappa \in A(i)$ is associated with a cost rate $\mathcal{C}_i\left(a_\kappa\right)$ in the the SMDP model. Typically the total cost incurred in a state is made of a fixed lump sum state-dependent cost, e.g, system initialization or switching cost, and a cost rate, which depends on current state, action taken, and possibly next state.

In the sensor node model, as energy consumption is the focus, costs are represented in each state by communication-related power consumption. The power consumption performance of the protocol governing the behavior of the node is then computed as long run expected accumulated cost per unit time. Formally the long run average cost under a policy f is defined as

$$G^f = \lim_{T\to\infty} E\left(\frac{C(T)}{T}\right), \tag{5.1}$$

where $E(\cdot)$ is the expectation operation and $C(T)$ is the accumulated cost over a period of time T. It has been shown in [80] that the long-run average cost can be computed as

$$G^f = \frac{\sum_{i\in\mathcal{I}}\sum_{a_\kappa\in A(i)}\hat{\pi}_i p_i(a_\kappa)c_i(a_\kappa)}{\sum_{i\in\mathcal{I}}\sum_{a_\kappa\in A(i)}\hat{\pi}_i p_i(a_\kappa)E_i(a_\kappa)}. \tag{5.2}$$

In reference to Section 5.2.2, the state transition probability of the embedded Markov process $\{X_\ell\}$ is $P_{ij} = \sum_{a_\kappa \in A(i)} p_i(a_\kappa) p_i(a_\kappa, j)$. The state probability $\hat{\pi}_i$ is computed using $\hat{\pi}\,[P_{ij}] = \hat{\pi}$, where $\sum_i \hat{\pi}_i = 1$.

5.4 Power Consumption Model for Sensor Nodes

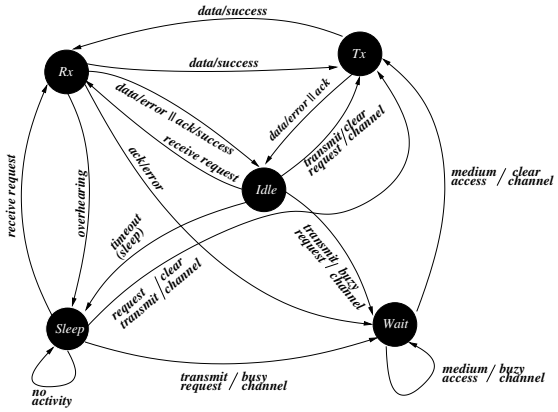


Figure 5.1 Schematic of the embedded Markov chain representing the nodes transition between the 5 states. Each state determines certain level of power consumption.

5.4.1 Network Topology and Traffic Model

Data collection and aggregation in a primary goals in many sensor network applications [81]. A network in a circular location space with randomly uniform distributed nodes is considered. The single data sink is located at the center of the space, which receives all sensed data and is typically a data processing center. All nodes generate packets according to a homogeneous Poisson process. In addition, each node also routes the traffic from its neighboring nodes towards the direction of the sink. The wireless channel is assumed ideal for capturing the effect of channel access conflict. In other words, transmission failure or link error is assumed to occur as a result of collision only. For tractability of the analysis, the node is modeled as a single customer service. Until the

currently waiting or lost (backlogged) packet is successfully (re)transmitted, new arrivals are discarded. Additional buffering consideration will be validated separately through simulation. The net new packet arrival rate at each node is assumed Poisson and is denoted by λ_n. An approximate amount of relay traffic that a node handles will depend on the packet forwarding used.

Traffic forwarding scheme: For the analysis, a simplified forwarding scheme, named *random forwarding*, is used where the field nodes send data packets towards the sink by randomly selecting a forward direction neighbor (FDN) as a relaying node in a multi-hop fashion. This simple forwarding approach provides a natural way of traffic load distribution as well as a tractable way to study the effect of the network multi-hop dynamics on the node control model.

Assume there are N nodes, of which $N-1$ are data sources. The nodes are denoted by IDs 0 to $N-1$, the sink node being node 0. Denote d_{xy} as the distance between nodes x and y (see Figure 5.2), and $d_{0x} \triangleq d_x$ to represent the distance of node x to the sink.

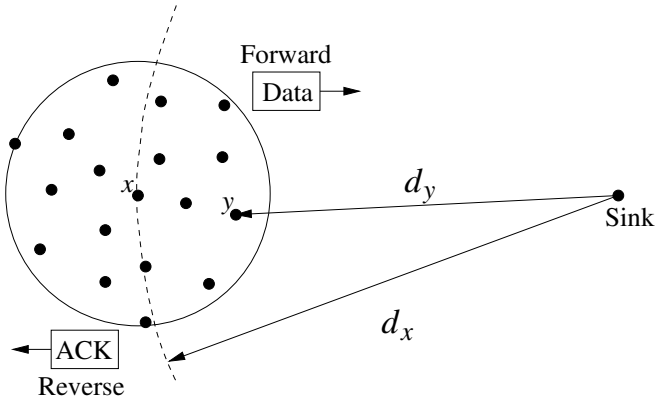


Figure 5.2 Data packets are sent towards the forward direction neighbors. ACKs are sent towards the reverse direction neighbors.

All nodes transmit at nominal power and have isotropic radio communication range r. The neighborhood relationships are defined as follows:

(i) Node y is said to be a neighbor of node x (and vice versa) if and only if $d_{xy} \leq r$;

(ii) Node y is said to be a forward direction neighbor (FDN) of node x if the following conditions are satisfied:

$$d_{xy} \leq r \text{ and } d_y \leq d_x$$

$\mathcal{N}_{\text{FDN}}(x)$ denotes the set of all FDNs of node x and $|\mathcal{N}_{\text{FDN}}(x)|$ the number of FDNs of node x.

(iii) Node x is said to be a reverse direction neighbor (RDN) of node y if the following conditions are satisfied:

$$d_{yx} \leq r, \text{ and } d_x > d_y$$

Similarly, $\mathcal{N}_{\text{RDN}}(y)$ denotes the set of all RDNs of node y and $|\mathcal{N}_{\text{FDN}}(y)|$ the number of RDNs of node y.

Note that (ii) and (iii) imply that directional (forward and reverse) neighborhoods are not meaningful for nodes at a distance smaller than r from the sink node. As defined, FDNs and RDNs are reciprocal, i.e.,

$$y \in \mathcal{N}_{\text{FDN}}(x) \Leftrightarrow x \in \mathcal{N}_{\text{RDN}}(y). \tag{5.3}$$

In the random forwarding scheme, neighborhood information is assumed to be available, and for relaying traffic a node selects the nodes in its FDN with equal probability. With these settings, data packets always travel in the forward direction while acknowledgment (ACK) packets travel in the reverse direction.

The relay traffic: The relay traffic at a node depends on both the physical node deployment and multi-hop forwarding strategy. Since the node distribution in the network is uniformly random and there is only one sink, the closer a node is to the sink, the greater the volume of relay traffic. Moreover, because new data packet arrivals are Poisson-distributed, the relay data traffic at a node is also Poisson-distributed with a distance-to-the-sink dependent rate $\lambda_r(d)$, where d is the distance of the node under consideration to the sink.

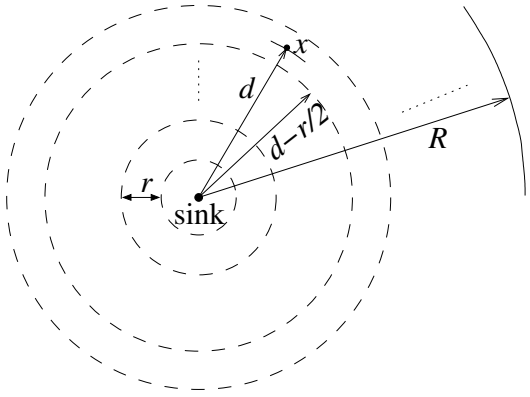


Figure 5.3 Schematic used for estimation of relay traffic.

To compute $\lambda_r(d)$, the circular network space of radius R is divided into concentric coronas of width r (radio range) around the sink, and a node, say x, at a distance d_x from the sink (see Figure 5.3) is considered. Let the average number of active neighbors of a node be n. The number of nodes in the corona at distance d_x is $\frac{2nd}{r}$. These nodes will have to carry the new traffic load $\frac{2nd\lambda_n}{r}$ plus the traffic relayed from outside rings $\frac{n\lambda_n}{r^2}\left[R^2-(d+r/2)^2\right]$. In the forwarding model considered, a packet could take multiple hops to cover the distance

r (radio range), which increases the traffic load of an intermediate node (say, the node x) further. Denoting the near field distance of a node as d_0, the number of hops within a range would vary within $[1, r/d_0]$, where it is assumed that r/d_0 is an integer, > 1. This within-range multi-hopping possibility approximately increases the locally generated traffic $\frac{2nd\lambda_n}{r}$ by a factor of $\frac{1}{2}\left(1+\frac{r}{2d_0}\right)$ and the relay traffic by a factor of $\frac{r}{2d_0}$. Thus, a node at a distance $(d \pm r/2)$ carries approximately $\frac{r}{2nd}\left\{\frac{2nd\lambda_n}{r}\cdot\frac{1}{2}\left(1+\frac{r}{2d_0}\right)+\frac{n\lambda_n}{r^2}\left[R^2-(d+r/2)^2\right]\cdot\frac{r}{2d_0}\right\}$ $= \lambda_n + \frac{\lambda_n}{2}\left(\frac{r}{2d_0}-1\right) + \frac{\lambda_n}{4dd_0}\left[R^2-(d+r/2)^2\right]$ data traffic. Hence, effectively, the approximate relay traffic carried by a node at a distance d (and in its neighborhood) is:

$$\lambda_r(d) \approx \frac{\lambda_n}{2}\left(\frac{r}{2d_0}-1\right)+\frac{\lambda_n}{4dd_0}\left[R^2-(d+r/2)^2\right] \tag{5.4}$$

In the modeling of energy consumption presented in the following section, for simplicity of the analysis, it is assumed that the traffic relayed by the neighbors of a node is approximately the same as that relayed by the node itself. The distance dependence term in the relay traffic rate is also omitted; the term λ_r is used, which implicitly means $\lambda_r(d)$. In Section 5.5, while computing numerical results, the distance dependence is taken into account to capture the energy consumption patterns of different nodes.

5.4.2 Receive State Analysis

When a node is in the *receive* (*Rx*) state, three alternative events are possible: it is either receiving data, or receiving ACK, or overhearing communication from neighboring nodes. After successfully receiving a data packet, the node immediately sends an ACK; otherwise it simply idles. If an ACK is received, the node goes to *idle* state. When a node overhears a transmission intended to other, it listens to the transmission long enough to determine that the packet is not destined to itself and then switches to the *sleep* state.

The state transitions described above (also see Figure 5.4) outline the decision rule followed by the virtual controller in the *Rx* state. The set of allowable actions in the *Rx* state is therefore

$$A(\text{Rx}) = \{\text{receive data, receive ACK, overhearing}\}.$$

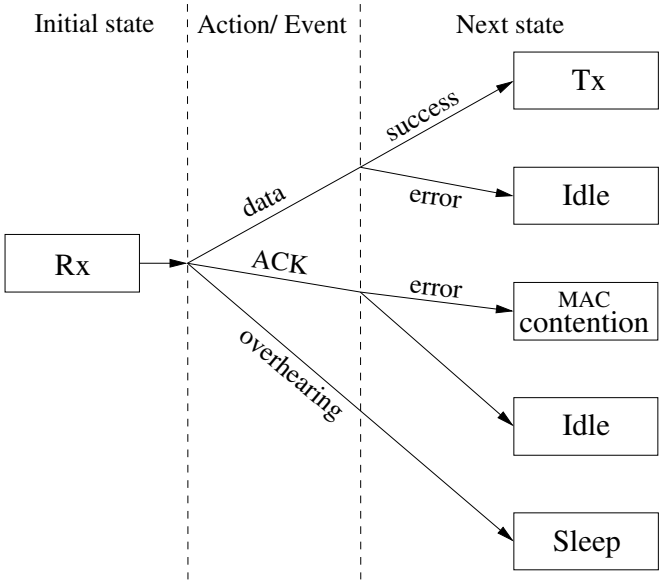


Figure 5.4 Decision rule in the *receive* (*Rx*) state.

Action/Event selection probability The conditional probabilities of *receiving data*, *receiving ACK*, and *overhearing*, given that the node is in the *Rx* state are computed. These probabilities depend on the node's neighborhood density (average number of local neighbors) and traffic load, and the network's multi-hop communication strategy.

In reference to Figure 5.2, assume that all nodes have the same average number of FDNs, $|\mathcal{N}_{\text{FDN}}(y)| = |\mathcal{N}_{\text{FDN}}(x)| = n_1$, and RDNs, $|\mathcal{N}_{\text{RDN}}(y)| = |\mathcal{N}_{\text{RDN}}(x)| = n_2$. Thus, the average number of local neighbors of a node, $|N(y)| = |N(x)| = n = n_1 + n_2$. By equation (5.3), the node y is a potential relay of x if $y \in \mathcal{N}_{\text{FDN}}(x)$ (or equivalently if $x \in \mathcal{N}_{\text{RDN}}(y)$). Consequently, $\mathcal{N}_{\text{RDN}}(y)$ represents the set of n_2 nodes that may chose node y as a relaying node. Also, because in random forwarding each node requesting a relay service evenly balances its traffic among its n_1 FDNs, $1/n_1$ of the forward direction traffic of node x is destined to y. Thus, a node picks up on average per unit time:

- Transmissions from its n_2 RDN
  - $n_2(\lambda_n + \lambda_r)$ data, of which $\frac{n_2}{n_1}(\lambda_n + \lambda_r)$ are destined to x,
  - $n_2\lambda_r$ ACK, of which none is destined to x.
- Transmissions from its n_1 FDN
  - $n_1(\lambda_n + \lambda_r)$ data, of which none is destined to x,
  - $n_1\lambda_r$ ACK, of which $\frac{n_1}{n_2}\lambda_r$ is destined to x.

From the above observations, the probabilities of receiving a data packet, an ACK, or an overheard packet at a node, while it is in the *Rx* state, can be obtained, respectively, as:

$$p_{rx}(\text{data}) = \frac{n_2\left(\lambda_n + \lambda_r\right)}{n_1\left(n_1 + n_2\right)\left(\lambda_n + 2\lambda_r\right)} \tag{5.5}$$

$$p_{rx}(\text{ack}) = \frac{n_1\lambda_r}{n_2\left(n_1 + n_2\right)\left(\lambda_n + 2\lambda_r\right)} \tag{5.6}$$

$$p_{rx}(\text{oh}) = 1 - \frac{{n_2}^2\lambda_n + \left({n_1}^2 + {n_2}^2\right)\lambda_r}{{n_1}^2{n_2}^2\left(n_1 + n_2\right)\left(\lambda_n + 2\lambda_r\right)} \tag{5.7}$$

For further simplification, let $n_1 = n_2$, i.e., assume symmetricity of forwarding directions, which is approximately the case when the distance to the sink is large and the transmission range $r \ll R$ (see Figure 5.2). This gives the simplified probability expressions:

$$p_{rx}(\text{data}) = \frac{\lambda_n + \lambda_r}{n\left(\lambda_n + 2\lambda_r\right)} \tag{5.8}$$

$$p_{rx}(\text{ack}) = \frac{\lambda_r}{n\left(\lambda_n + 2\lambda_r\right)} \tag{5.9}$$

$$p_{rx}(\text{oh}) = 1 - \frac{1}{n} \tag{5.10}$$

where $n = n_1 + n_2$ is the average number of local neighbors. Note that equation (5.10) verifies the intuition that with symmetricity assumption and random forwarding strategy, the intended transmission from a node (say node y) to a neighbor (say node x) in its *Rx* state is with probability $1/n$.

Sojourn times To determine the sojourn time distribution in the *Rx* state, recall that the SMDP model allows the time spent in a particular state to have a general distribution, and can depend on the action chosen by the decision-maker (virtual controller) as well as the next state of the system. In capturing the average cost (energy spent) per unit time, particularly interest is given to the averages of sojourn times.

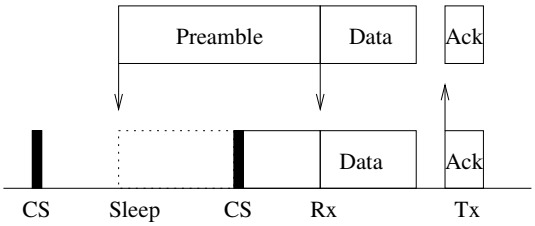


Figure 5.5 Data transmit and receive process

Receiving data: For successfully receiving a data packet, a node must remain in *Rx* state during the time necessary to receive the entire packet. Packets of fixed-size L_{data} are considered for all nodes. As depicted in Figure 5.5, it is also assumed that every data packet reception is preceded by a part or whole of a preamble (for waking up the receiving node). It follows that the sojourn time in *Rx* state while successfully receiving data is uniformly distributed between $\left(\frac{L_{data}}{C}\right)$ and $\left(\frac{L_{data}+L_{preamble}}{C}\right)$, where $l_{preamble}$ is the fix length of the preamble, and C the data rate of the wireless channel. Recalling that every successful data packet reception is followed by an ACK transmission (i.e., the next state is *Tx*), and denoting by $E_i(a_\kappa, j)$ the average sojourn time in state i if the action a_κ is taken in the present and the next state of the system is j,

$$E_{rx}(\text{data}, \textit{Tx}) = \frac{L_{data} + L_{preamble}/2}{C}. \tag{5.11}$$

Also, because the data packet error is assumed with due forwarding inability or channel access conflict, and noting that the error checksum is appended at the trailer of the data packet, the sojourn time in *Rx* state while a data packet reception fails is uniformly distributed between 0 and $\left(\frac{L_{data}+L_{preamble}}{C}\right)$. Thus,

$$E_{rx}(\text{data}, \textit{idle}) = \frac{L_{data} + L_{preamble}}{2C}. \tag{5.12}$$

Receiving ACK: ACKs are in general of smaller size and are sent without preamble. It is aslo assumed that every data packet is acknowledged separately. Piggybacking is not used because, in the type of network under consideration, data and ACKs travel in opposite directions. The distribution of sojourn time in *Rx* state when ACK is being received is therefore the time needed to successfully or unsuccessfully receive the ACK, which is respectively given by

$$E_{rx}(\text{ack}, \textit{idle}) = \frac{L_{ack}}{C}, \tag{5.13}$$

for successfully receiving an ACK, where L_{ack} is the length of an ACK, and

$$E_{rx}(\text{ack}, \textit{MAC}) = \frac{L_{ack}}{2C}, \tag{5.14}$$

in case of failure to successfully receive the ACK.

Overhearing: Overhearing occurs when a node is engaged in receiving transmission that is not intended for itself. It is assumed that the time necessary for a node to detect

overhearing is at least the time required to receive the header of the data, ACK. Let L_h the length of the header of both data and ACK packets. In case the overheard transmission is a data packet, preamble time also needs to be taken into account. Particularly, the sojourn time is exactly $\frac{L_h}{C}$ when the overheard transmission is an ACK, and it is uniformly distributed between $\left(\frac{L_h}{C}\right)$ and $\left(\frac{L_h+L_{preamble}}{C}\right)$ if the overheard transmission is a data packet. Following the description in Section 5.4.2, it can be shown that in unit time, a node overhears

- $\left(n_1 + n_2 - \frac{n_2}{n_1}\right)(\lambda_n + \lambda_r)$ data packet transmissions, and
- $\left(n_1 + n_2 - \frac{n_1}{n_2}\right)\lambda_r$ ACK transmissions.

Therefore, the conditional probabilities of overhearing data and ACKs are, respectively, given by

$$\Pr\{\text{data}|\text{oh}\} = \frac{\left(n_1 + n_2 - \frac{n_2}{n_1}\right)(\lambda_n + \lambda_r)}{\left(n_1 + n_2 - \frac{n_2}{n_1}\right)(\lambda_n + \lambda_r) + \left(n_1 + n_2 - \frac{n_1}{n_2}\right)\lambda_r} \tag{5.15a}$$

$$\Pr\{\text{ack}|\text{oh}\} = \frac{\left(n_1 + n_2 - \frac{n_1}{n_2}\right)\lambda_r}{\left(n_1 + n_2 - \frac{n_2}{n_1}\right)(\lambda_n + \lambda_r) + \left(n_1 + n_2 - \frac{n_1}{n_2}\right)\lambda_r} \tag{5.15b}$$

The average sojourn time would then be obtained as:

$$E_{rx}(\text{oh, } \textit{sleep}) = \frac{L_h}{C} \cdot \Pr\{\text{ack}|\text{oh}\} + \frac{L_h + L_{preamble}/2}{C} \cdot \Pr\{\text{data}|\text{oh}\} \tag{5.16}$$

Making the same simplifying assumption as in equations (5.8)-(5.10) (i.e., by letting $n_1 = n_2$) the from (5.15):

$$\Pr\{\text{data}|\text{oh}\} = \frac{\lambda_n + \lambda_r}{\lambda_n + 2\lambda_r} \tag{5.17a}$$

$$\Pr\{\text{ack}|\text{oh}\} = \frac{\lambda_r}{\lambda_n + 2\lambda_r} \tag{5.17b}$$

can be obtained, which gives the average sojourn time

$$E_{rx}(\text{oh, } \textit{sleep}) = \frac{\lambda_n + \lambda_r}{\lambda_n + 2\lambda_r}\frac{L_h}{C} + \frac{\lambda_r}{\lambda_n + 2\lambda_r}\frac{L_h + L_{preamble}/2}{C} \tag{5.18}$$

Transition probabilities Referring to Figure 5.4, the transition probabilities from *Rx* state to other state of the node model can be computed.

Data packets: After successful reception of a data packet, nodes switch to Tx state to transmit an ACK. Consequently, the probability of going to *Tx* after data packet reception corresponds to the probability that the packet is received successfully. To capture only the MAC-related power consumption, it is assumed packet reception fails only in the case of collision.

Consider that node x is receiving data from node y. For simplification, further consider that only nodes in the neighborhood of node x, but not in that of node y, can cause a collision with the packet from node y. In the worst case, the number of interfering nodes is

$$\max\{|N(x)| - |N(x) \cap N(y)|\} = \frac{n}{\pi}\left(\frac{2\pi + 3\sqrt{3}}{6}\right)$$

There transition probability into Tx follows

$$p_{rx}(\text{data, } \textit{Tx}) = 1 - p_{collision}^{data}$$

where

$$p_{collision}^{data} = e^{\left\{-\frac{n}{\pi}\left(\frac{2\pi+3\sqrt{3}}{6}\right)(\lambda_n+\lambda_r)E_{rx}(\text{data, } \textit{Tx})\right\}} \tag{5.19}$$

and $E_{rx}(\text{data, } \textit{Tx})$ is the time spent by node x to receive the entire data packet including preamble. Correspondingly, the probability of going into *idle* state is

$$p_{rx}(\text{data, } \textit{idle}) = p_{collision}^{data} \tag{5.20}$$

ACKs: First note that a node expecting/receiving ACK has been a data packet transmitter at the previous state. If the ACK is received successfully, the node simply idles. However, unsuccessful/missing ACK represents the failure of the previously sent data packet. The node switches to MAC contention mode for retransmission.

The same analysis made above to compute the probability of collision for data packets can be made to compute the probability of collision of ACKs.

$$p_{collision}^{ack} = e^{\left\{-\frac{n}{\pi}\left(\frac{2\pi+3\sqrt{3}}{6}\right)(\lambda_n+\lambda_r)E_{rx}(\text{ack, } \textit{MAC})\right\}} \tag{5.21}$$

where $E_{rx}(\text{ack, } \textit{MAC})$ is the transmission time of an ACK packet. The probability of switching to *MAC contention* and the probability of switching to *idle* are given by

$$p_{rx}(\text{ack, } \textit{MAC}) = p_{collision}^{ack} \tag{5.22}$$

$$p_{rx}(\text{ack, } \textit{idle}) = 1 - p_{collision}^{ack} \tag{5.23}$$

Overhearing: The outcome of transmission overhearing is invariably switching to $sleep$ state, i.e.,

$$p_{rx}\left(\text{oh, } \textit{sleep}\right) = 1. \tag{5.24}$$

5.4.3 Idle State Analysis

Idle listening corresponds to the state in which a node listens actively to potentially transmit or receive packets, and it is a major source of consumption of unnecessary energy in sensor networks. To reduce energy consumption due to idle listening, timeout policy is often used [78] to automatically switch the controlled system into a lower power-consuming state.

In *idle* state, the events and the corresponding allowable actions of the decision-maker are (see also Figure 5.6):

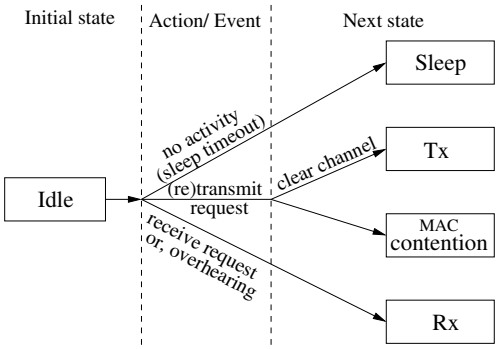


Figure 5.6 Decision rule in the *idle* state.

- The node goes to *sleep* after no activity for a timeout threshold t_{th}.
- The node goes to *Tx* or MAC contention, depending on the channel condition after an interruption of idle period because of new or backlogged data packet to be sent.
- The node goes to *Rx* after an interruption of idle period because of data packet to be received.

Below, the probability of each of these events, corresponding sojourn times, and associated costs are computed.

Action/Event selection probabilities First note that when a node x is in idle state, all transmissions and receptions from or to node x concern data packets, and no ACKs are involved. The reason is that whenever an ACK has to be transmitted or received, the node will either switch directly from *Tx* to *Rx* mode (to receive the ACK) or from *Rx* to *Tx* mode (to transmit the ACK). In both cases, no transition through the *idle* state is needed. In other words, when in idle state, a node is not required to transmit or receive an ACK. However, an idling node can overhear ACKs from its neighbors. Thus, the idle period of node x can only be interrupted due to (i) data packets to be transmitted by the node itself, (ii) data packets to be received or overheard from neighboring nodes, and (iii) overhearing ACKs. It follows that in the idling phase, the total rate of transmit attempt would be $\lambda_n+\lambda_r$, and the total rate of receive/overhear attempt would be $n(\lambda_n+\lambda_r)+n\lambda_r-(\lambda_n+\lambda_r) = (n-1)\lambda_n+(2n-1)\lambda_r$, where $n(\lambda_n + \lambda_r)$ represents the total number of data packets sent by n neighbors of node x, and $n\lambda_r$ is the total number of ACKs sent, of which $(\lambda_n + \lambda_r)$ represents the portion destined to node x.

Therefore, sleep timeout probability, i.e., the probability that there is no transmit, receive, or overhearing activity until timeout t_{th} occurs is given by

$$p_{idle}(\text{sleep timeout}) = e^{-n(\lambda_n+2\lambda_r)t_{th}}. \tag{5.25}$$

Correspondingly, transmit request probability, i.e., the probability of the idling period being interrupted by a transmit or retransmit request, is given by

$$p_{idle}(\text{tx request}) = \left(1 - e^{-n(\lambda_n+2\lambda_r)t_{th}}\right) \frac{\lambda_n + \lambda_r}{n\left(\lambda_n + 2\lambda_r\right)} \tag{5.26}$$

where $\left(1 - e^{-n(\lambda_n+2\lambda_r)t_{th}}\right)$ is the probability that at least one activity (transmit or receive request or overhearing) occurs before timeout, and $\frac{\lambda_n+\lambda_r}{n(\lambda_n+2\lambda_r)}$ represents the probability that a data transmit request occurs before receive request, or overhearing.

A receive request or overhearing probability during the idle period is

$$p_{idle}(\text{rx request or oh}) = \left(1 - e^{-(\lambda_n+2\lambda_r)t_{th}}\right) \frac{(n-1)\,\lambda_n + (2n-1)\,\lambda_r}{n\left(\lambda_n + 2\lambda_r\right)} \tag{5.27}$$

Sojourn time As in the other states, the sojourn time in the idle state depends on the actions chosen by the decision-maker.

Sleep timeout: In the event of sleep timeout, the node stays in idle mode for the time t_{th}. Therefore,

$$E_{idle}(\text{no activity, } \textit{sleep}) = t_{th} \tag{5.28}$$

Transmit/retransmit request: If the idle period is interrupted by a (re)transmit request for a new or backlogged packet, there could be two possible outcomes: going to *Tx* state or MAC contention (or carrier sensing (CS)) state. Since the (re)transmit requests arrival is a Poisson process with parameter $\lambda_n + \lambda_r$, the interarrival time is exponentially distributed; however it is truncated at the upper limit of t_{th}. The average sojourn time is therefore

$$\begin{aligned} E_{idle}(\text{tx request, } \textit{Tx}) &= E_{idle}(\text{tx request, } \textit{cs}) \\ &= \frac{1}{\lambda_n + \lambda_r} - e^{-(\lambda_n+\lambda_r)t_{th}} \left(t_{th} + \frac{1}{\lambda_n + \lambda_r} \right) \end{aligned} \tag{5.29}$$

Receive request or overhearing: Since λ_n and λ_r are Poisson-distributed, the interarrival times of receive requests and overhear signals are also exponentially distributed with parameter $(n-1)\lambda_n + (2n-1)\lambda_r \triangleq \lambda_{rx}$, but truncated at the upper limit of t_{th}. The corresponding average sojourn time is given by

$$E_{idle}(\text{rx request, } \textit{Rx}) = \frac{1}{\lambda_{rx}} - e^{-\lambda_{rx} t_{th}} \left(t_{th} + \frac{1}{\lambda_{rx}} \right)$$

(5.30)

Transition probabilities The transition probabilities out of *idle* state are now computed. Referring to Figure 5.6, the transition probabilities in the event of sleep timeout or receive request are straight forward:

$$p_{idle}\,(\text{sleep timeout, } \textit{sleep}) = 1 \tag{5.31}$$

$$p_{idle}\,(\text{rx request, } \textit{Rx}) = 1 \tag{5.32}$$

When a node in $idle$ state needs to transmit its own data, the wireless channel is first sampled according to the MAC protocol, and the node only proceeds to *Tx* state if the channel is found idle. The probability of finding the channel idle is now computed.

Since all data packets are of the same size, given amount of time is required to complete the full cycle for sending data packets including preamble, actual data packet, and the subsequent (eventual) ACK (see Figure 5.5). In the worst case, every transmission attempt from a node causes the wireless channel to be unavailable for other neighboring nodes for a time $t_{TxCycle} = \frac{L_{preamble}+L_{data}+L_{ACK}}{C}$. The probability for a node (say x) to find

the channel idle is the probability that none of the n neighboring nodes of node x initiated transmission in the preceding $t_{TxCycle}$ interval of time. It follows that

$$p_{idle}\,(\text{tx request, } \textit{Tx}) = e^{-n(\lambda_n+\lambda_r)t_{TxCycle}} \tag{5.33}$$

$$p_{idle}\,(\text{tx request, } \textit{Tx}) = 1 - e^{-n(\lambda_n+\lambda_r)t_{TxCycle}} \tag{5.34}$$

5.4.4 Transmit State Analysis

When a node is in *Tx* mode, two alternative events are possible: the node is transmitting either data or an ACK. After successfully sending a data packet, the transmitter immediately switches to *Rx* mode to receive an ACK. If the data packet transmission is in error (i.e., if no ACK is received), the transmitter goes to *idle* state from *Tx* state. A node, at the receiver end, goes to *Tx* state to send an ACK and then becomes *idle* after receiving the packet successfully. The state transition rules are shown in Figure 5.7.

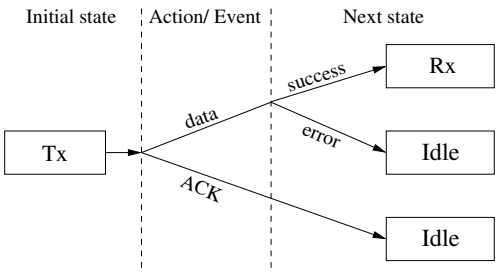


Figure 5.7 Decision rule in *transmit* (*Tx*) state.

Note that, even if the data packet is received successfully at the receiver, it could be retransmitted due to ACK failure (handled by the *Rx* state, shown in Figure 5.4). On

the other hand, if the data is not received successfully at the receiver, the data packet is backlogged at the transmitter and from the idle state it goes through the MAC contention phase for its retransmission attempt (as depicted in Figure 5.6).

Action/Event selection probability In the transmit state, the net data packet arrival rate is $\lambda_n + \lambda_r$ and the ACK arrival rate is λ_r. Therefore, the event selection probabilities are, respectively,

$$p_{tx}(\text{data}) = \frac{\lambda_n + \lambda_r}{\lambda_n + 2\lambda_r} \tag{5.35}$$

$$p_{tx}(\text{ack}) = \frac{\lambda_r}{\lambda_n + 2\lambda_r}. \tag{5.36}$$

Transition Probabilities Here, the same analysis is made as in the case of Rx. When Data packets are involved, all packet collision are assumed to occur only on the receiver end because nodes in the neighborhood of the transmitter refrain from transmitting if they sense any activity on the channel. The probability of packet failure due to collision is the same as in equation (5.19).

$$p_{tx}(\text{data, } \textit{Rx}) = 1 - p_{collision}^{data} \tag{5.37}$$

$$p_{tx}(\text{data, } \textit{idle}) = 1 - p_{collision}^{data} \tag{5.38}$$

$$p_{tx}(\text{ack, } \textit{idle}) = 1 \tag{5.39}$$

Sojourn Times As in the *Rx* state, following the similar arguments for sojourn time evaluation,

$$E_{tx}(\text{data, } Rx) = \frac{L_{data} + L_{preamble}}{C} \tag{5.40}$$

$$E_{tx}(\text{data, } idle) = \frac{L_{data} + L_{preamble} + L_{ack}}{C} \tag{5.41}$$

$$E_{tx}(\text{ack, } idle) = \frac{L_{ack}}{C}. \tag{5.42}$$

5.4.5 Sleep State Analysis

In the *sleep* state, the radio transceiver of the sensor node is switched off to avoid unnecessary idle listening and minimize energy consumption. During the sleep state, if a transmit request arrives, the node immediately wakes up to transmit the data packet. Also, without any transmit request, a sleeping node periodically wakes up and samples the wireless channel for activity to maintain the network connectivity. If the node detects any activity, it switches to *Rx* mode. Otherwise, it goes to the next sleep cycle. A fixed sleeping length t_{sleep} is considered for all nodes. However, the occurrence of sleep state of a node is independent of the others', so that the probability of all nodes in the same neighborhood being in sleep mode is negligible. Thus, in the sleep state, three alternative events can occur (see Figure 5.8):

- No activity: the sleeping period finishes without any transmit or receive request,
- Transmit request: immediate wake-up if a new packet arrives during the sleeping period, and

- Receive request: wake-up after the current sleep period if a receive request is made from a neighboring node.

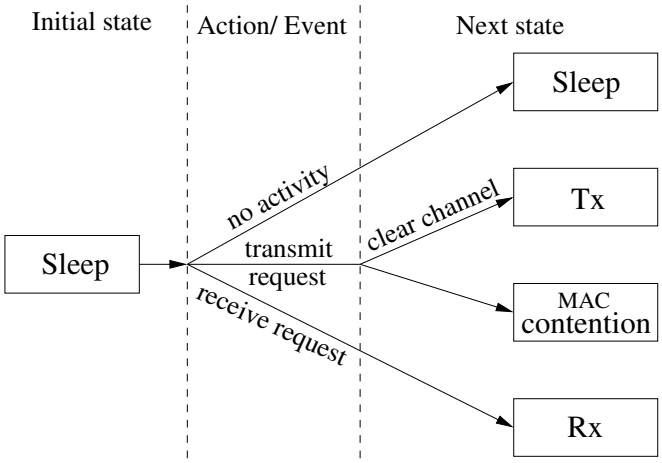


Figure 5.8 Decision rule in *sleep* state.

Action/Event Selection Probabilities The probability of no activity in all n neighbors of the sleeping node and no new packet arrival during a sleep period t_{sleep} is given by

$$p_{sleep}(\text{no activity}) = e^{-(n+1)(\lambda_n+\lambda_r)t_{sleep}}. \tag{5.43}$$

A transmit request arrives with probability

$$p_{sleep}(\text{tx request}) = 1 - e^{-(\lambda_n+\lambda_r)t_{sleep}}. \tag{5.44}$$

Likewise, the probability of a sleeping node receiving a relay request from one of its neighbors, but no new packet arriving, is given by

$$p_{sleep}(\text{rx request}) =$$

$$e^{-(\lambda_n+\lambda_r)t_{sleep}}\left(1-e^{-n(\lambda_n+\lambda_r)t_{sleep}}\right) \tag{5.45}$$

Sojourn Time Since a receive request is attended to only after the current sleep period, the sojourn time due to receive request is the same as that due to no activity, which is

$$E_{sleep}(\text{rx request, } Rx) = E_{sleep}(\text{no activity, } sleep)$$

$$= t_{sleep} \tag{5.46}$$

A transmit request originating at the sleeping node leads to two possibilities: going to the *Tx* state (with clear channel) or the MAC contention (or CS) state (with channel busy). In either case, sojourn time is the same, as in the case of an idle state:

$$E_{sleep}(\text{tx request, } Tx) = E_{sleep}(\text{tx request, } CS)$$
$$= \frac{1}{\lambda_n+\lambda_r} - e^{-(\lambda_n+\lambda_r)t_{sleep}}\left(t_{sleep}+\frac{1}{\lambda_n+\lambda_r}\right) \tag{5.47}$$

The transitions probabilities in $sleep$ state are similar to that of $Idle$ state (compare Figure 5.8 and Fig. 5.6).

5.4.6 MAC Contention

MAC contention resolution is used to make efficient use of the shared wireless medium among nodes in the same vicinity. In the MAC scheme considered, a node samples the wireless channel whenever it desires to transmit data, and backs-off for a random time if the channel is not idle; otherwise the node transmits its data (see Figure 5.9). For simplicity,

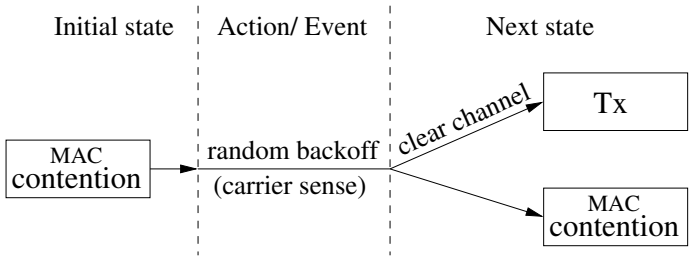


Figure 5.9 Decision rule in MAC contention state.

it is assumed that the back-off time to be continuously uniformly distributed between 0 and $t_{back-off}$. In the decision model, MAC contention (or physical CS) corresponds to a single action *back-off* selected with probability 1 whenever the node is in this state.

$$p_{CS}(\text{back-off}) = 1 \tag{5.48}$$

At the end of each *back-off* period, the node samples the wireless channel again and transmits its data if the channel is found to be idle; otherwise it backs-off for a random time.

The transition probabilities for *MAC contention* can be derived from that of the *idle* state (compare Figure 5.9 and Fig. 5.6).

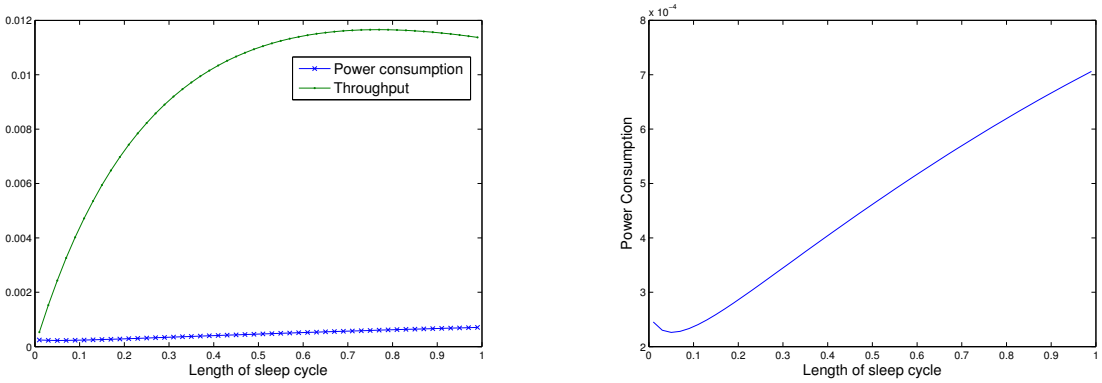
5.5 Numerical Results

In this section, an application of the power consumption model is presented, and results are obtained through numerical computation. The proposed model of sensor nodes as controlled stochastic dynamic systems is applied to study the effect on communication-related power consumption of the node sleep cycle. As protocol designers, communication scientists and engineers often face optimal design and control problems. The goal in optimal design is to find a fixed optimal value of a system parameter, while in dynamic control, the goal is to search for strategies to adjust the system parameter in order to achieve optimal system operation. In both cases, an in-depth understanding of the system behavior under complex protocols is needed.

To illustrate how the model can be used to gain in-depth understanding of the complex behavior of sensor nodes under the combined actions of MAC protocols and power management policies, the effect of sleeping period on the node level long-run power consumption has be studies under low and high traffic loads. In addition, based on the traffic model and assumptions from Section 5.2, an energy consumption map of the sensor field at low and high traffic loads is obtained. In the numerical studies, $R = 100$, and the sink is located at the center. Radio range is $r = 10$, and the near field distance $d_0 = 1$. Node density (i.e., average number of neighbors n of a node) and new traffic generation rate λ_n is varied, and λ_n in turn modifies the relay traffic $\lambda_r(d)$.

Fig. 5.10(a) shows both the energy consumption and throughput performance of a node x at a fixed distance ($d = 95$) from the sink. As all nodes in the same neighborhood

sleep longer, the packet collision probability drops and the throughput of node increases. In turn, the overall energy consumption increases.



(a) Power consumption (b) throughput

Figure 5.10 Long-run average power consumption and throughput performance of a node at $d = 95$ from the sink. $\lambda_n = 0.01$.

Fig. 5.10(b) shows the energy consumption per node as the length of sleep cycle is increased. The results are counter-intuitive: the long-run average power consumption increases as a node sleeps longer in each sleep cycle. Because the node is modeled as a single-customer service, it is important in this case to compare the node's power consumption performance to its utilization. A measure of throughput is defined as the product of the probability that a node is in transmit state, the probability that the node is transmitting data, and the probability that the data are successfully received at the receiver end. As defined, this throughput performance factor represents the effective utilization of the sensor node, i.e., the fraction of time the node spends successfully sending data packets.

The previous observation, however, only holds true for low traffic load. Refer to Figure 5.11, where energy and throughput are calculated for a node a distance $d = 40$ to the sink. Since the traffic load is high in the neighborhood of node y, collisions are more likely to occur, and nodes (mostly backlogged) will spend more time in a lower-consuming

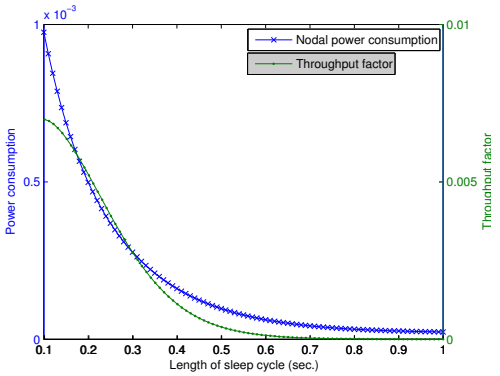


Figure 5.11 Effect of sleep cycle on energy consumption and throughput at $d = 40m$. $\lambda_n = 0.01$.

state (sleep and MAC contention). As a consequence, an increase in sleep time decreases the energy consumption, at the cost of lower throughput performance.

The above observation suggests the existence of two regions of operating conditions for the sensor nodes:

- one desirable region, in which an increase in traffic load results in an increase of energy consumption, and
- one undesirable region, in which an increase in traffic load may result in a decrease in overall energy consumption because nodes, being overwhelmed by traffic, block most of the incoming packets.

Although nodes are modeled without consideration for buffering, the behavior of nodes in a practical implementation will be similar to buffered nodes' behavior in the undesirable region if nodes are overwhelmed beyond their storage capacity. In this case, most incoming traffic is blocked, and nodes spend most of the time in contention mode. Because energy consumption in the contention state is lower than that required for packet transmission and reception, there is an apparent net decrease in long-run energy

consumption. However, because throughput also decreases at the same time, most of the energy is wasted resolving contention.

The numerical results show that undesirable operational region can be avoided by selecting node density and system parameters as to keep collision probability low for all nodes at all times. This condition is met in Figure 5.12, where the nodes, when subject

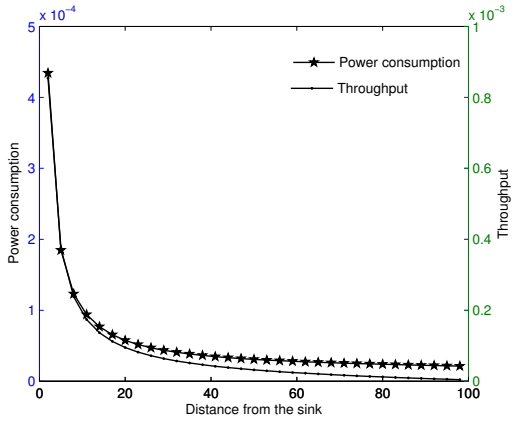


Figure 5.12 Energy consumption map of the sensor field with low traffic load, $\lambda_n = 0.0001$ and $n = 15$.

to very low traffic load, consume energy that mostly follows the traffic pattern. However,

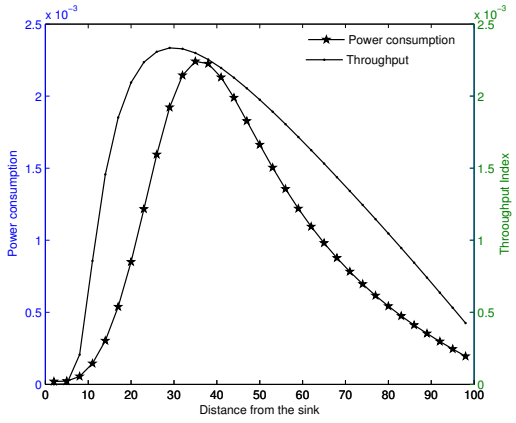


Figure 5.13 Energy consumption map of the sensor field with high traffic load. $\lambda_n = 0.01$, $n = 15$.

Figure 5.13 shows a network scenario where nodes around the sink are overwhelmed with

traffic and operate in the undesirable region. Figure 5.12 also indicates that for a given single sink location, a suitable distance-dependent node density could be derived that would enable all nodes in the network to spend the same amount of energy.

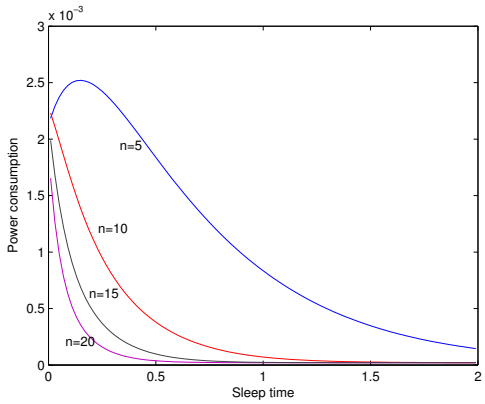


Figure 5.14 Power power consumption at various sleep time. $\lambda_n = 0.1$, $\doteq 40$.

The plots in Figure 5.14 show power consumption behavior with sleep time at different node densities. Since the new traffic arrival rate is kept at a constant high ($\lambda_n = 0.1$), at very low sleep time, low node density cannot handle the traffic properly and shows the undesired convex region, as in Figure 5.13. This example indicates again that, for a desirable network behavior, there is an allowable traffic load with a given uniformly random node density.

5.6 Conclusions

An analytic model was developed for capturing the power consumption of a peer-to-peer sensor node under a given communication protocol. A sensor node was modeled as a controlled stochastic dynamic system, and the model was formulated as a SMDP, wherein a node's states, sojourn times, and the transition probabilities are controlled by a virtual node

controller, that is specific to a given communication protocol set. The energy efficiency of a protocol was measured as a long-run average cost per unit time.

The developed model assumed a preamble sampling-based low power communication protocol as an underlying MAC protocol and a random forwarding strategy that probabilistically selects a forward direction node for multi-hop-forwarding a packet. The numerical results demonstrated that the model can be used effectively as a tool for optimizing different protocol parameters to achieve energy-efficient communication goals.

CHAPTER 6

CONCLUSION AND FUTURE WORK

This work addressed challenges faced by the choice of intermediary relaying node during multi-hop packet forwarding in error wireless networks. Because of the unreliable nature of the wireless medium, relay nodes selected only based on location information (greedy geographic forwarding), also tend to be more unreliable, leading to increased retransmission, throughput drop and energy inefficiency. Also, centralised forwarding decision made at the transmitter side has the additional burden of collecting all require decision parameters increasing the risk of using inaccurate and outdated information for packet forwarding. To remedy these drawbacks, packet forwarding in multi-hop wireless network was treated fundamentally as a multi-criteria decision problem where wireless link quality information can be added to location information for the choice of best relay.

First, to alleviate the burden of information gathering at a single decision point, receiver-side relay election was proposed as a distributed election among potential relay candidates. It was shown that with judicious mapping of decision parameter (e.g. hop-progress) onto time delays used to rank the nodes, a distributed election can be held with minimum collusion and time delay. Next, multi-criteria receiver-side relay election framework for multi-hop relaying in ad hoc networks was considered. Via intuitive reasoning and examples it was shown qualitatively the importance of finding optimum weighted relay election/selection criteria. A generalized cost metric in the form of multi-parameter mapping function has been proposed and used to investigate optimal tradeoff between greedy forwarding and link quality. It has been shown that a much

better network performance in terms of total energy consumption for successful end-to-end routing can be achieved via judicious selection of the weighting parameter that optimally trades off between greediness and link quality. The multi-criteria mapping function is quite general and can also be applicable to transmitter-side relay selection process. Lastly, the possibility of multi-path forwarding was evaluated.

REFERENCES

[1] J. Han, W. Zhao, and M. Zheng, "An analytical model on unbalanced energy consumption in large scale wireless sensor network," in *Measuring Technology and Mechatronics Automation (ICMTMA), 2011 Third International Conference on*, vol. 1, jan. 2011, pp. 375 –378.

[2] H. Idoudi, N. Hmili, and L. Saidane, "Energy efficient cross-layer architecture for wireless sensor networks," in *Mediterranean Microwave Symposium (MMS), 2011 11th*, sept. 2011, pp. 126 –129.

[3] J. Yang, D. Zhang, and Y. Zhang, "An energy-efficient data gathering protocol for wireless sensor networks," in *Computer and Information Science, 2009. ICIS 2009. Eighth IEEE/ACIS International Conference on*, Jun. 2009, pp. 780 –785.

[4] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393 – 422, 2002.

[5] M. Baysan, K. Sarac, R. Chandrasekaran, and S. Bereg, "A polynomial time solution to minimum forwarding set problem in wireless networks under unit disk coverage model," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 20, no. 7, pp. 913 –924, july 2009.

[6] P. Kaewprapha, J. Li, and N. Puttarak, "Network localization on unit disk graphs," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, dec. 2011, pp. 1 –5.

[7] E. Lebhar and Z. Lotker, "Unit disk graph and physical interference model: Putting pieces together," in *Parallel Distributed Processing, 2009. IPDPS 2009. IEEE International Symposium on*, may 2009, pp. 1 –8.

[8] S. Funke, A. Kesselman, U. Meyer, and M. Segal, "A simple improved distributed algorithm for minimum cds in unit disk graphs," in *Wireless And Mobile Computing, Networking And Communications, 2005. (WiMob'2005), IEEE International Conference on*, vol. 2, aug. 2005, pp. 220 – 223 Vol. 2.

[9] W.-J. Liu and K.-T. Feng, "Greedy routing with anti-void traversal for wireless sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 7, pp. 910 –922, july 2009.

[10] K. Egoh and S. De, "Priority-based receiver-side relay election in wireless ad hoc sensors networks," in *Proc. MILCOM*, Vancouver, BC, Canada, Jul. 2006, pp. 1177–1182.

[11] B. Karp and H. T. Kung, "Gpsr: Greedy perimeter stateless routing for wireless networks," in *Proc. ACM MOBICOM*, Aug. 2000, pp. 243–254.

[12] M. Zorzi and R. Rao, "Geographic random forwarding (geraf) for ad hoc and sensor networks: Multihop performance," *IEEE Trans. Mobile Computing*, vol. 2, no. 4.

[13] H. Takagi and L. Kleinrock, "On hop count and euclidean distance in greedy forwarding in wireless ad hoc networks,," *IEEE Commun. Letters*, vol. 9, no. 11.

[14] R. P. Q. H. Q. H. Guoliang Xing, Chenyang Lu, "On greedy geographic routing algorithms in sensing-covered networks," in *In Proceedings of MobiHoc 04*, 2004, pp. 31–42.

[15] G. Xing, C. Lu, R. Pless, and Q. Huang, "Impact of sensing coverage on greedy geographic routing algorithms," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 17, no. 4, pp. 348 – 360, april 2006.

[16] K. Yu and S. Bu, "Loop-free greedy routing in ad hoc or sensor networks using muti-hop geographic information," in *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, vol. 3, Nov. 2009, pp. 328 –332.

[17] G. Tan and A.-M. Kermarrec, "Greedy geographic routing in large-scale sensor networks: A minimum network decomposition approach," *Networking, IEEE/ACM Transactions on*, vol. PP, no. 99, pp. 1 – 14, 2011.

[18] Q. Chen, S. Kanhere, and M. Hassan, "Analysis of per-node traffic load in multi-hop wireless sensor networks," *Wireless Communications, IEEE Transactions on*, vol. 8, no. 2, pp. 958 –967, feb. 2009.

[19] H. Frey and D. Gorgen, "Geographical cluster-based routing in sensing-covered networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 17, no. 9, pp. 899 –911, sept. 2006.

[20] A. Dvir and N. Carlsson, "Power-aware recovery for geographic routing," in *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, april 2009, pp. 1 –6.

[21] H. Frey and D. Gorgen, "Geographical cluster based routing in sensing-covered networks," in *Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference on*, june 2005, pp. 885 – 891.

[22] R. Shirani, M. St-Hilaire, T. Kunz, Y. Zhou, J. Li, and L. Lamont, "The performance of greedy geographic forwarding in unmanned aeronautical ad-hoc networks," in *Communication Networks and Services Research Conference (CNSR), 2011 Ninth Annual*, may 2011, pp. 161 –166.

[23] C. Qian and S. Lam, "Greedy distance vector routing," in *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, june 2011, pp. 857 –868.

[24] D. Tschopp, S. Diggavi, M. Grossglauser, and J. Widmer, "Robust geo-routing on embeddings of dynamic wireless networks," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, may 2007, pp. 1730 –1738.

[25] N. Arad and Y. Shavitt, "Minimizing recovery state in geographic ad hoc routing," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 2, pp. 203 –217, feb. 2009.

[26] K. Seada, A. Helmy, and R. Govindan, "On the effect of localization errors on geographic face routing in sensor networks," in *Information Processing in Sensor Networks, 2004. IPSN 2004. Third International Symposium on*, april 2004, pp. 71 – 80.

[27] P. Y. Xu and Z. L. Jun, "Virtual destination based geographic routing in ad hoc mobile networks," in *Wireless Communications, Networking and Mobile Computing, 2005. Proceedings. 2005 International Conference on*, vol. 2, sept. 2005, pp. 686 – 689.

[28] G. Zhao, X. Liu, and M.-T. Sun, "Anchor-based geographic routing for sensor networks using projection distance," in *Wireless Pervasive Computing, 2007. ISWPC '07. 2nd International Symposium on*, feb. 2007.

[29] B. Raja, N. Prabakaran, and V. Dhulipala, "Modified gpsr based optimal routing algorithm for reliable communication in wsns," in *Devices and Communications (ICDeCom), 2011 International Conference on*, feb. 2011, pp. 1 –5.

[30] Y. Sun, Q. Jiang, and M. Singhal, "An edge-constrained localized delaunay graph for geographic routing in mobile ad hoc and sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 9, no. 4, pp. 479 –490, april 2010.

[31] L. Zhao, H. Zhu, Y. Xu, and X. Li, "Legr: A load-balanced and energy-efficient geographic routing for lossy wireless sensor network," in *Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007. 3rd International Conference on*, dec. 2007, pp. 119 –124.

[32] Y.-B. Wang, T.-Y. Wu, W.-T. Lee, and C.-H. Ke, "A novel geographic routing strategy over vanet," in *Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on*, april 2010, pp. 873 –879.

[33] A. H. K. Seada, M. Zuniga and B. Krishnamachari, "Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks," Baltimore, MD, 2004, pp. 108–121.

[34] B. B. S. Lee and S. Banerjee, "Efficient geographic routing in multihop wireless networks," in *Proc. ACM MobiHoc*, Urbana-Champaign, IL, May 2005, pp. 230–241.

[35] M. R. Souryal and N. Moayeri, "Channel-adaptive relaying in mobile ad hoc networks with fading," in *Proc. IEEE SECON*, Santa Clara, CA, 2005.

[36] J. W. H. Fubler and M. Kasemann, "Contention-based forwarding for mobile ad hoc networks," *Elsevier Ad Hoc Networks*, vol. 1, no. 4.

[37] K. Egoh and S. De, "A multi-criteria receiver-side relay election approach in wireless ad hoc networks," in *Proc. MILCOM*, Washington, DC, Oct. 2006.

[38] H. Takagi and L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals,," *IEEE Trans. Commun.*, vol. COM-32, no. 3.

[39] J. W. M. Mauve and H. Hartenstein, "A surveyon position-based routing in mobile ad hoc sensor networks," *IEEE Network Mag.*, vol. 15, pp. 30–39, 2001.

[40] F. J. R. Tanbourgi, H. Jakel, "Increasing the one-hop progress of nearest neighbor forwarding," *IEEE Communications Letters*, vol. 15, no. 21 month = Jan, year = 2011, issn = 1687-1472, pages = 64–66,.

[41] K. Dai, Q. Wang, W. Xu, C. Han, M. Xiong, and Y. Cheng, "Link-aware geographic routing in wsn," in *Web Society (SWS), 2010 IEEE 2nd Symposium on*, aug. 2010, pp. 231 –236.

[42] Q. Zhao and L. Tong, "Opportunistic carrier sensing for energy-efficient information retrieval in sensor networks," *EURASIP J. Wirel. Commun. Netw.*, vol. 2005, no. 2, pp. 231–241, Apr. 2005.

[43] X. L. C. Yi, P. Wan and O. Frieder, "Fault tolerant sensor networks with bernoulli nodes,," in *Proc. IEEE WCNC*, New Orleans, LA, Mar. 2003.

[44] R. K. S.C. Zhang, F.I. Koprulu and D. Jones, "Feasibility analysis of stochastic sensor networks,," in *Proc. of the 1st IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, 2004, pp. 4–7.

[45] C.-T. Kuo and C.-K. Liang, "A meshed multipath routing protocol in mobile ad hoc networks," in *Proc. International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'06)*, 2006, pp. 306–310.

[46] S. R. S. S. A. Rao, C. Papadimitrou and I. Stoica, "Geographic routing without location information,," in *Proc. ACM MOBICOM*, San Diego, CA, 2003, pp. 96–108.

[47] C. S. J. Camp, J. Robinson and E. Knightly, "Measurement driven deployment of a two-tier urban mesh access network," in *Proceedings of the 4th international conference on Mobile systems, applications and services*, New York, NY, USA, 2006, pp. 96–109.

[48] J. P. R. Draves and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proceedings of the 10th annual international conference on*, ser. MobiCom '04, New York, NY, USA, 2004, pp. 114–128.

[49] D. N. N.S Nandiraju and D. Agrawal, "Multipath routing in wireless mesh networks," in *Proc. Mobile Adhoc and Sensor Systems (MASS)*, Vancouver, BC, Oct. 2006, pp. 741–746.

[50] P. Gupta and P. Kumar, "The capacity of wireless networks," *Information Theory, IEEE Transactions on*, vol. 46, no. 2, pp. 388 –404, mar 2000.

[51] D. Manikantan Shila, Y. Cheng, and T. Anjali, "Capacity of cooperative wireless networks using multiple channels," in *Communications (ICC), 2010 IEEE International Conference on*, may 2010, pp. 1 –5.

[52] K. Gomadam, V. Cadambe, and S. Jafar, "Approaching the capacity of wireless networks through distributed interference alignment," in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, 30 2008-dec. 4 2008, pp. 1 –6.

[53] Y. Nebat, R. Cruz, and S. Bhardwaj, "The capacity of wireless networks in nonergodic random fading," *Information Theory, IEEE Transactions on*, vol. 55, no. 6, pp. 2478 –2493, june 2009.

[54] J. Kirkebo, D. Gesbert, and S. Kiani, "Maximizing the capacity of wireless networks using multi-cell access schemes," in *Signal Processing Advances in Wireless Communications, 2006. SPAWC '06. IEEE 7th Workshop on*, july 2006, pp. 1 –5.

[55] T. Wang, G. Leus, and L. Huang, "Ranging energy optimization for robust sensor positioning based on semidefinite programming," *Signal Processing, IEEE Transactions on*, vol. 57, no. 12, pp. 4777 –4787, dec. 2009.

[56] S. Wolfram, *A New Kind of Science*. Wolfram Media, 2002.

[57] H. Lee and A. Keshavarzian, "Towards energy-optimal and reliable data collection via collision-free scheduling in wireless sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, april 2008, pp. 2029 –2037.

[58] H. Zhang and H. Shen, "Energy-efficient beaconless geographic routing in wireless sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 21, no. 6, pp. 881 –896, june 2010.

[59] O. Younis and S. Fahmy, "Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 3, no. 4, pp. 366 – 379, oct.-dec. 2004.

[60] W. Li, M. Bandai, and T. Watanabe, "Tradeoffs among delay, energy and accuracy of partial data aggregation in wireless sensor networks," in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, april 2010, pp. 917 –924.

[61] A. Chamam and S. Pierre, "On the planning of wireless sensor networks: Energy-efficient clustering under the joint routing and coverage constraint," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 8, pp. 1077 –1086, aug. 2009.

[62] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Optimizing sensor networks in the energy-latency-density design space," *Mobile Computing, IEEE Transactions on*, vol. 1, no. 1, pp. 70 – 80, jan-mar 2002.

[63] B. Otal, L. Alonso, and C. Verikoukis, "Highly reliable energy-saving mac for wireless body sensor networks in healthcare systems," *Selected Areas in Communications, IEEE Journal on*, vol. 27, no. 4, pp. 553 –565, may 2009.

[64] W. Fang, F. Liu, F. Yang, L. Shu, and S. Nishio, "Energy-efficient cooperative communication for data transmission in wireless sensor networks," *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 4, pp. 2185 –2192, november 2010.

[65] J. Luo, L. Jiang, and C. He, "Cross-layer optimization for energy-timeliness tradeoff in tdma based sensor networks," in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, 30 2008-dec. 4 2008, pp. 1 –5.

[66] A. Reinhardt, D. Christin, M. Hollick, and R. Steinmetz, "On the energy efficiency of lossless data compression in wireless sensor networks," in *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on*, oct. 2009, pp. 873 –880.

[67] A. Sinha and A. Chandrakasan, "Dynamic power management in wireless sensor networks," *Design Test of Computers, IEEE*, vol. 18, no. 2, pp. 62 –74, mar/apr 2001.

[68] J. H. W. Ye and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *Proc. IEEE INFOCOM*, New York, NY, 2002, pp. 1567–1576.

[69] P. Basu and J. Redi, "Effect of overhearing transmissions on energy efficiency in dense sensor networks," in *Proc. ACM IPSN*, Berkeley, CA, Apr. 2004, pp. 196–204.

[70] J. Hill and D. Culler, "Mica: a wireless platform for deeply embedded networks," *IEEE Micro*, vol. 22, no. 6, pp. 12–24, Nov./Dec. 2002.

[71] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks," in *Proc. IEEE ICC*, vol. 5, Apr. 2002, pp. 3418 – 3423.

[72] J. H. J. Polastre and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. ACM SenSys*, Baltimore, MD, Nov. 2004.

[73] Q. W. Q. Qiu and M. Pedram, "Stochastic modeling of a power-managed system – construction and optimization," *IEEE Trans. Computer Aided Design of Integrated Circuits and Systems*, vol. 20, no. 10, pp. 1200–1217, Oct. 2001.

[74] P. Rong and M. Pedram, "Battery-aware power management based on markovian decision process," in *Proc. IEEE/ACM Intl. Conf. Computer-aided design*, San Jose, CA, Nov. 2002, pp. 707–713.

[75] C.-F. Chiasserini and M. Garetto, "Modeling the performance of wireless sensor networks," in *Proc. IEEE INFOCOM*, vol. 1, Hong Kong, China, Mar. 2004, pp. 1–12.

[76] C. P.-R. Z. Shelby and J. Haapola, "Energy optimization in multihop wireless embedded and sensor networks," in *Proc. IEEE Intl. Symp. PIMRC*, vol. 1, Barcelona, Spain, Sep. 2004, pp. 221–225.

[77] Chipcon, *CC2420 Datasheet, http//wwwchipcon.com/files/CC2420_Data_Sheet_1_2.pdf, June 2004*.

[78] P. G. T. Simunic, L. Benini and G. D. Micheli, "Dynamic power management for portable systems," in *Proc. ACM MOBICOM*, Boston, MA, Aug. 2000, pp. 11–19.

[79] T. Simunic and S. Boyd, "Managing power consumption in networks on chip," in *Proc. ACM Conf. Design, Automation and Test*, Paris, France, Mar. 2002, pp. 110–117.

[80] V. G. Kulkarni, *Modeling, Analysis, Design, and Control of Stochastic Systems*. Springer-Verlag, New York, 1999.

[81] X. Xu, X.-Y. Li, X. Mao, S. Tang, and S. Wang, "A delay-efficient algorithm for data aggregation in multihop wireless sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 22, no. 1, pp. 163 –175, Jan 2011.