Multiterminal Relay Networks: Performance Bounds, Protocol Design, and Channel Coding Strategies

by

Bin Zhao

Dissertation submitted to the
College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Electrical Engineering

Matthew C. Valenti, Ph.D., Chair
Lawrence A. Hornak, Ph.D.
Roy S. Nutter, Jr., Ph.D.
Daryl Reynolds, Ph.D.
Sherman D. Riemenschneider, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2004

Keywords: Ad Hoc Networks, Cross-Layer Design, Relay Networks, Cooperative Diversity, Adaptive Relaying, Hybrid ARQ Based Geographic Relaying, Distributed Turbo Codes

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Abstract

Multiterminal Relay Networks: Performance Bounds, Protocol Design, and Channel Coding Strategies

by

Bin Zhao
Doctor of Philosophy in Electrical Engineering

West Virginia University

Matthew C. Valenti, Ph.D., Chair

The combination of silicon scaling and energy-efficient multi-terminal packet radio technology will soon allow low power devices to be embedded virtually everywhere, enabling a wide range of revolutionary applications that will radically change the way that people and devices interact with their environments. The broader societal impact of embedded networks will be profound, enabling new services to benefit almost all aspects of life. Given current trends in the advancement of technology, wireless networks of limited utility, scale, and lifetime are possible without much further research. However, in order to engineer useful embedded wireless networks with long lifetimes and massive scale required for many applications, new analytical tools and approaches to protocol design that reflect recent perspectives on wireless networking are necessary.

The major objective of this dissertation is to characterize the fundamental performance bounds and devise an integrated approach to the design, analysis, and implementation of energy efficient cross-layer protocols for wireless embedded networks under realistic constraints. The focus of the study is on a general class of embedded wireless networks that are decomposed into clusters of several low cost radio devices including a source, a destination, and one or more relays. The message propagation mechanism of each cluster is modelled as a rate constrained relay network in which signaling is over a random phase block interference channel, and transmissions from the various nodes are non-coherent. Numerical analysis indicates that even in relay networks under small rate constraints, e.g. $M = 2$ orthogonal transmissions, significant energy savings are achievable by implementing *distributed* spatial diversity via adaptive or nonadaptive relaying. For relay networks under large rate constraints, we propose energy-efficient relaying protocols that jointly perform cooperative diversity, hybrid-ARQ retransmission, and routing, first for time-invariant networks to exploit a better energy-throughput tradeoff over multihop or direct transmission, and then for time-varying networks to fully implement the time and spatial diversity with energy constrained devices. Unlike multihop, where a network-layer protocol is needed to explicitly select a message route through the network a priori, relaying will adaptively find the best 'path' and will tend to bypass relays that are continually in an outage, thus power/range control becomes less important in relay networks. On the other hand, as relaying requires many more devices than multihop to listen to each broadcast, its energy efficiency benefit begins to diminish due to non-negligible energy cost to receive a transmission. Therefore, to avoid excessive receiver energy dissipation in large scale networks, the coverage area and optimal cluster size of relaying need to be carefully defined. Finally, we propose simple coding strategies inspired by the turbo principle is proposed to approach the information theoretic limits of the constrained relay networks under block fading.

Acknowledgments

First of all, I would like to thank Dr. Matthew C. Valenti who has been a terrific advisor and under whom I was a research assistant at West Virginia University. His insight and invaluable suggestions helped shape this dissertation and guide my research. I would also like to thank my committee members Dr. Lawrence A. Hornak, Dr. Roy S. Nutter, Jr., Dr. Daryl Reynolds, and Dr. Sherman D. Riemenschneider for their help and valuable suggestions on my research and dissertation.

Finally, on a personal note, I would like to thank my wife Bin Feng for all her support and understanding throughout the process and to whom I dedicate this dissertation.

Contents

List of Figures

Chapter 1

Introduction

1.1 Objectives and Significance

The combination of silicon scaling and energy-efficient multi-terminal packet radio technology will soon allow low power devices to be embedded virtually everywhere, enabling a wide range of revolutionary applications that will radically change the way that people and devices interact with their environments [1]. The broader societal impact of embedded networks will be profound, enabling new services of benefit to industry, transportation, medicine, science, agriculture, national security, disaster relief, and the environment protection. Recently, the National Research Council identified embedded networks of sensors and actuators as a research area of great national importance [2], and the IEEE 802.15 wireless personal area network (WPAN) task group 4 is currently standardizing ultra-low power networks [3, 4].

Embedded networks of sensors and actuators belong to an important subclass of ad hoc wireless networks which is identified as a collection of wireless mobile nodes self-configured to form a network without the aid of any established infrastructure [5]. A major distinctive feature of these networks is the necessity of multihop relaying before the message reaches its destination. In the design and analysis of ad hoc networks with high efficiency, significant research effort has been driven toward two extreme ends of distinctive applications. At one extreme is the research that relies on complicated signal processing and detection techniques to approach the information theoretic limits of wireless networks which are typically characterized by their high spectral efficiency applications and expensive network devices. Network information theory is the major field of study for those network applications. At the other extreme is the research on embedded networks that pursues energy efficient techniques for low cost network devices and massive scale deployment. In particular, for most applications, it is inconvenient or even impossible to replace their batteries, thus the devices must operate at extremely low power. Each device must expend a certain amount of energy to transmit or receive a message, and when its energy reserves are depleted, it must either be recharged or else leave the network. This places a particular emphasis on the issue of energy consumption and its impact on the methodology used to design and analyze wireless protocols [6].

When engineering energy-efficient networks, we should make a clear distinction between networks that treat energy as a cost function (but renewable) and networks that treat it as a hard constraint (non-renewable). Although hard energy constraints are not inherent to all networks, energy constrained networks comprise an important subclass of embedded networks and are the enabling technology for the most critical and exciting applications up to date. It is quite evident that even in networks supplied with very large or renewable energy sources, energy consumption is a still a big concern to achieve cost-effective operation. Theoretically, for embedded networks with renewable energy resources, energy efficiency is achievable at the price of spectrum efficiency, i.e. spread spectrum or ultra-wideband signaling. However, when energy constraints are imposed on embedded networks, although energy efficiency is important, some issues are at least as important as, if not more important than, energy efficiency. For instance, the longevity of networks and the total number of data bits delivered. In general, for networks with energy constraints, higher energy efficiency does not necessarily result in longer network lifetime or more delivered data. Another open question is that for a network to only last for certain amount of operation time, what should be the most critical performance metric? The throughput, or the total number of data bits delivered? [6]. In fact, when dealing with networks with finite energy reserves, the conventional design and analysis methodologies are no longer valid, since the design objectives for energy constrained network are highly application specific. Likewise, it is equally difficult to come up with an absolute performance metric to evaluate the network operation. A more detailed discussion about the impact of finite energy reserves on embedded network design will be postponed to the end of this chapter.

It is evident that the design for energy-efficient embedded networks could always benefit from employing advanced technologies in modern communications and wireless networking, however the device cost and low energy restrictions constantly discourages such employment. Therefore, most research on embedded networks has resorted to layer coupling and device cooperation for energy conservation. It is well known that layering is a form of hierarchical modularity central to the design and implementation of communication networks. However, it has become obvious that separating network functions into layers and characterizing general network architectures according to the 7-layer open systems interconnection (OSI) reference model, has, to some extent, turned out be the "original sin" in networking [6]. Although it facilitates initial comprehension of network operation and simplifies the complexity of network design, the layered structure results in suboptimal solutions, since it inevitably ignores the intrinsic interdependency among multiple layers. The initiation of layer-coupling and device cooperation in the study of ad hoc wireless networks has opened up a new horizon that holds huge potentials for energy savings compared to the conventional design of isolated protocol layers.

The main objective of this dissertation is to characterize the fundamental performance bounds of embedded networks and devise an integrated approach to the design, analysis, and implementation of energy efficient cross-layer protocols for wireless embedded networks under realistic constraints. The focus of the study is on a general class of wireless embedded networks that are decomposed into clusters of several low cost radio devices including a source, a destination, and one or more relays. Each cluster works cooperatively to convey information from source to destination. In chapter 2, the message propagation mechanism of each cluster is modelled as a random access relay network in which signaling is over a random phase block interference channel, and transmission from the various nodes are non-coherent. Chapter 3 studies the random access relay network under the constraint that at most two blocks are transmitted. The maximum number of transmission blocks allowed per message is usually defined as the rate constraint M. In particular, as $M = 2$, the relay network reduces to rate constrained orthogonal relay channels where orthogonality is achieved through Time Division Multiplexing (TDM) or Frequency Division Multiplexing (FDM). Closed form bounds on the performance of constrained relay channels are derived in terms of channel capacity and outage probability. Several new adaptive cross-layer relaying protocols are proposed to improve the relay channel performance. Numerical analysis indicates that even under small rate constraints significant energy savings are possible by implementing spatial diversity in the relay channel through device cooperation.

The study of rate constrained relay channels is later extended to constrained relay networks with larger rate constraints and multiple relay nodes in chapter 4. In particular, hybrid Automatic Repeat reQuest (ARQ) is employed to guide the message transmission within the random access relay network. The energy saving through hybrid-ARQ is apparent, since relays only need to transmit when the destination does not correctly decode the source message, however message delay becomes a major concern in the practical implementation. Essentially, the protocol design boils down to exploiting the fundamental tradeoff between energy efficiency and throughput in relay networks. Several relaying protocols are proposed and their performance investigated in terms of throughput, delay, and energy-efficiency. It turns out that by exploiting *distributed* spatial diversity, relaying is able to achieve better energy-throughput tradeoff than either multihop or direct transmission.

Because of the finite energy reserve in network devices, aggressive power-off strategies are commonly employed to conserve energy in embedded networks. In order to collect spatial diversity in networks with sleeping nodes, in chapter 5, we further incorporate MAC and routing design into the relaying protocols. In particular, if nodes know their own position and messages are addressed by location, then it is possible to use this geographic information to guide the routing mechanism. This is the underlying concept of the protocol that we propose and term Hybrid ARq-Based Intra-cluster GEographically-informed Relaying (HARBINGER). More specifically, this new protocol utilizes geographic information to jointly perform physical-layer cooperative diversity, data-link-layer hybrid-ARQ retransmission, and network layer relaying/routing. The analysis of HARBINGER in AWGN channel generalizes Geographic Random Forwarding (GeRaF) [7], which corresponds to the specific case that rate constraint $M = 1$. Numerical results indicate that even without cooperative diversity, HARBINGER is especially beneficial in lower density networks due to its significant reduction of message delay under relatively large rate constraint. Accordingly, in a densely deployed network, a smaller duty-cycle sleep schedule could be used for network devices with HARBINGER, thereby increasing the useful lifetime of embedded networks. Alternatively, for the same sleep schedule, HARBINGER allows reduced end-to-end delay compared to GeRaF. Several versions of the HARBINGER protocol with considerably different behavior have been proposed to meet different requirements of network applications. To simplify the analysis of HARBINGER, each network device is assumed to flush their memory after each successful message transmission, but this memory flushing reduces the cooperative diversity/relaying benefit in densely deployed networks. Therefore without memory flushing, advantages of HARBINGER in embedded networks over multihop routing should be more evident in terms of energy efficiency and throughput, especially for energy constrained networks over block fading. Finally, in chapter 6, a simple coding strategy inspired by the turbo principle is proposed and shown to approach the information theoretic limit of the constrained relay networks under small rate constraints. For networks with multiple relay terminals and larger rate constraint, distributed multiple turbo coding was proposed and shown to achieve significant diversity and coding gain in relay networks under Rayleigh fading.

1.2 Design Challenges of Ad Hoc Networks

Although the focus of this study is on a special subclass of ad hoc networks suitable for low-cost devices, massive scale deployment and energy efficient operation, a general overview of the design challenges of ad hoc networks is necessary to provide insights into the design of embedded networks. Those who wish to design ad hoc networks are faced with a dilemma. In particular, if the network design assumes maximum flexibility to support many applications, it will be difficult to tailor the network to different application requirements. This will likely result in poor performance for some applications especially those with high rate requirement or stringent delay constraints. On the other hand, if the network is tailored to a few specific applications, the designer must predict in advance what these killer applications will be–a risky proposition. Ideally an ad hoc wireless network must be sufficiently flexible to support many different applications while adapting its performance to the given set of applications in operation at any given time [5]. The cross-layer design approach provides the flexibility, while still able to tailor protocol design to the energy constraints in the network devices.

In the following sections, we will briefly inspect some important techniques and issues in wireless networking and finally discuss the methodology used in cross-layer design.

1.2.1 Modulation/Coding

Modulation and coding are physical layer techniques that at a minimum must convey information over the channel, but if implemented effectively can be used to improve the link quality. While the physical layer primarily focus on implementation details of physical channel communication, its interface to the data link layer appears as a virtual bit pipe for sequence transmission with designated Signal to Noise Ratio (SNR). This SNR metric is directly related to the maximum achievable rate of transmission in this virtual bit pipe, commonly referred to as the channel capacity [8]. A practical concern is how to approach capacity with reasonable complexity. Shannon uses infinitely long block length, purely random code structure, as well as maximum likelihood decoding to achieve channel capacity. Obviously, Shannon's approach is more of a mathematical existence proof rather than a practical solution. On the one hand, with joint optimization of modulation and coding, trellis coded modulation (TCM) [9] improves error performance of data links without sacrificing data rate or requiring more bandwidth. However, TCM is highly sensitive to carrier-phase tracking errors, and thus should be applied with caution to fading channels. On the other hand, advanced channel coding techniques approach channel capacity with serial and parallel concatenation codes [10] [11], i.e. turbo codes, and codes on graph, i.e. LDPC [12]. The distinctive characteristics of these codes are the application of code concatenation to achieve a pseudo-random code structure and iterative processing to approximate maximum likelihood detection with complexity that is linear in the block length. The same design and detection principles could be applied in modern communication systems where the cascaded functional blocks could be conceptually considered as generalized code concatenation, thus iterative detection technique could be readily applied. Despite its exceptional ability to save transmit energy, the application of such signal detection techniques should be handled with care, since turbo processing requires much more signal processing power than most other detection algorithms due to its iterative nature.

1.2.2 Multiple Antennas

It has been widely understood that the quality of wireless links can be significantly improved by exploiting the spatial diversity available to multiple transmit and receive antennas systems (MIMO). Spatial diversity is possible whenever there is spatial separation between multiple devices such that the diversity branches experience uncorrelated fading. Due to the existence of spatial diversity, MIMO channels have been shown to achieve significantly higher spectral efficiency and diversity gain than conventional single antenna systems. For instance, a beamforming effect at the transmitter could be achieved by sending the same information through multiple transmit antennas (complicated signal processing could be used to facilitate coherent transmission). MIMO can improve channel capacity even without transmitter beamforming if a space-time code is used. In general, the maximum diversity gain of mn could be achieved in a m-transmit n-receive antennas system to combat detrimental fading effects. Meanwhile, in MIMO channel, independent fading paths increase the degrees of freedom available for wireless communication [13] and create multiple parallel spatial channels through which independent information could be transmitted to increase the spectral efficiency of the channel. This effect is formally known as spatial multiplexing. There exists a fundamental tradeoff between diversity gain and multiplexing gain in MIMO systems [14].

Despite the potential that MIMO systems possess to improve physical layer performance, for some applications such as embedded networks, antenna arrays and complicated signal processing algorithms are too cumbersome and expensive to be employed at each device. However, the low device costs associated with these networks allows them to be blanketed with a dense deployment of devices. Therefore, the performance of embedded networks can be improved by exploiting the intrinsic spatial diversity, namely *distributed* spatial diversity, due to the presence of multiple devices at different locations. Relaying [15] offers a flexible and cost-effective solution to collect spatial diversity. Based on layer coupling and device cooperation, relaying assembles multiple network devices to form a virtual “antenna array”, thus conventional MIMO technologies can then be readily applied to collect spatial diversity. Note that although MIMO saves transmit power, it consumes significantly more processing power due to its complicated signal processing algorithms. Therefore we need to carefully examine if MIMO technologies result in net energy savings in the physical layer design of ad hoc wireless network.

1.2.3 Adaptive Resource Allocation

In the energy efficient design of ad hoc wireless networks, adaptive resource allocation is a critical design issue especially for physical and link layer designs. Adaptive resource allocation adapts transmit power, data rate, modulation, and channel coding to either compensate or take advantage of the channel fluctuation to satisfy certain application requirements. Research results [16] indicate only modest adaptation in system parameters is necessary to achieve optimal performance. Further note that channel retransmission schemes that are integrated with forward error correction coding (FEC) constitute an important subclass of adaptive resource allocation techniques. In particular, type II hybrid Automatic Repeat reQuest (ARQ) protocols use diversity combining or code combining of the retransmitted code bits or packets to significantly increase throughput as well as energy efficiency. Essentially, hybrid-ARQ is an automatic rate/power adaptation scheme to mitigate the link fluctuation. It is automatic because no link estimation is necessary at the transmitter. A practical concern in the retransmission protocol is the frame size, since each transmission contains a constant packet overhead. The selection of optimal frame size involves a tradeoff between throughput and energy efficiency.

Power control is an important adaptive resource allocation technique that impacts the design at multiple layers. For instance, in physical and link layers, power control is used to maximize data throughput and channel capacity through water-filling [17] to take advantage of SINR fluctuation in the fading channel. It can also be used to reduce co-channel interference and meet hard delay constraints. However, its influence goes far beyond that. Random access protocols, which are typically implemented within the MAC sublayer, can be made more efficient and distributed by using power control [18]. In the network layer, power control could further influence routing by affecting the local neighborhood and network connectivity. Thus, power control essentially is a cross-layer issue in designing ad hoc networks.

1.2.4 Medium Access Control

Medium access control could be considered in general as a resource allocation problem too. It mainly deals with how different users efficiently share the common resource: the frequency spectrum. This problem eventually boils down to two practical issues: channelization and channel allocation. As for channelization, there has been well-developed solutions such as time division, frequency division and code division to divide available spectrum into orthogonal channels so that through appropriate channel assignment, interference within the local neighborhood could be minimized and the channel resources could be reused. In general, it is hard to determine which method is better, especially when the reuse factor has been taken into account. However, recent research indicates that pulsed operation of each cell can increase battery lifetime [19]. This result implies that time division might be more energy efficient than other channelization schemes for low-energy network implementation.

The design objective for channel allocation is to maximize the system throughput while at the same time avoiding collision as much as possible, since each packet collision indicates a certain amount of energy waste. The effectiveness of different techniques proposed to meet the design objective heavily relies on the network traffic types. For instance, some users have a lot of data to transmit within a very short time period (bursty traffic), while other users may have more continuous data stream. The traffic in ad hoc wireless networks is more or less a mixture of both. The random access protocol ALOHA [20] and its variants have been proven to be effective in bursty network traffic. With ALOHA, users have to contend for channel access for every packet. In particular, each user bursts whenever they have data to transmit. If more than two users transmit simultaneously, a collision occurs. ALOHA resolves collisions by forcing those users involved in the collision to wait a random amount of time before attempting to burst again. When ALOHA is reinforced by forward error correction codes [21], both throughput and energy efficiency are significantly improved, since packet retrieval is still possible even under collision. Recently, innovative research has been applied game theory [22] and dynamic programming into ALOHA schemes so that random access could become more flexible and energy-aware. Alternatively, carrier sense multiple access (CSMA) [23] requires that users sense the channel before transmission. However, the efficiency of CSMA is affected by hidden terminal problems when each node can only hear its immediate neighbor in the network and exposed terminal problems where some nodes may not transmit even if no collision would have occurred. Techniques such as four-way handshaking and busy tone transmission have been developed to mitigate both problems. On the other hand, when network traffic is composed of continuous data stream, multiple access turns out to be a scheduling problem where data transmission is connection-oriented. In particular, once a deterministic peer-to-peer channel has been captured by an individual user, it will be preserved until the user has finished transmitting all packets. ALOHA does not work well for continuous data streams due to its frequent resource contention scheme. Although centralized scheduling is common in cellular systems, in ad hoc networks, scheduling is carried out in a distributed manner and has been proved to be an NP-hard problem [24]. Recently, robust MAC protocols [25] have been proposed that incorporate scheduling into ALOHA to handle the coexistence of different traffic types in ad hoc networks.

1.2.5 Routing

The efficiency of routing algorithms is heavily influenced by network topology. In ad hoc networks, network topology is a rather volatile, commonly affected by channel fluctuation, network mobility, and adaptive resource allocation techniques. To increase the system tractability, a majority of research assumes static network structure. In general, routing algorithms [26] could be categorized into three major types: flooding, proactive routing and reactive routing. Flooding protocols simply broadcast message packets, and should any node correctly detect the message, that node will again broadcast the message to the rest of the network. This process continues until the destination receives the message. Although flooding is simple and robust, it causes packet duplication and unnecessary interference to other nodes' transmissions, resulting in spectral and energy inefficiency. Proactive routing is proposed to overcome the drawback of flooding in a static network. With proactive routing, a routing table is computed before routing takes place. The routing tables could either be generated locally (distributed manner) or optimized globally (centralized manner). In time-varying networks, frequent updates of routing tables take up a fair amount of traffic overhead. It is evident that distributed routing is suboptimal, however it involves less traffic overhead since information only needs to be exchanged within the local neighborhood. Therefore, distributed routing is especially desirable in time varying networks. Finally, routing tables could be calculated on-demand (reactive routing) where message paths/routes are only generated right before the source has packets to transmit. Reactive routing is able to compute globally optimal routes, however it causes significant routing delay due to its on-site path retrieval mechanism. In general, decent routing algorithms for ad hoc networks should be able to balance message delay and routing efficiency for different applications.

Another important issue is single hop vs. multi-hop routing. Although multi-hop could increase capacity [27], the best strategy for achieving the corresponding multi-hop gain with distributed algorithms is still an open problem. From an energy dissipation perspective, although multiple short hops tend to save transmit energy due to non-linear path loss, it increases receiver energy dissipation and increases message delay. In general, it is rather vague whether multi-hop is more advantageous than single hop in the sense of interference as well as efficiency.

1.2.6 The Impact of Finite Energy Reserve

Many applications of ad hoc networks rely on nonrenewable energy sources that significantly complicates network design and implementation. In particular, Shannon theory based on average or peak power constraints and error-free communications, is no longer valid to predict the fundamental limits of these networks. A suitable framework comes from Gallager's pioneering work [28] where he defined the reliable communication under a hard transmit energy constraint in terms of capacity per unit energy. This definition ensures that for all rates below the capacity per unit energy, error probability decreases exponentially with the total energy, although it will not be asymptotically small for finite energy channels. As opposed to the asymptotically error-free communication in Shannon theory, Gallager's results focused more on energy-efficient operation where errors are inevitable due to finite energy constraints. To achieve this theoretic limit, Gallager uses an unlimited number of degrees of freedom per transmitted bit. This suggests wideband communication or the use of many symbols per bits. The same concept is further explored in [29] and [30]. These research results serve as an important guideline in the design and analysis of energy constrained networks. In particular, with finite energy reserve, the total number of bits each device could transmit is predetermined by hard energy constraints. Then efficient allocation of these bits to components at different layers is a complex multi-dimensional optimization problem to which cross-layer design eventually becomes the last resort. In addition, scheduling at the MAC layer becomes more complicated in the sense that each bit in the energy constrained system needs to transmit over a long period of time in order to maximize the channel capacity, thus the ideal admission process becomes unrealistically long. In systems with multiple users, the tradeoff between transmission energy and scheduling time needs to be carefully considered.

1.2.7 Cross-Layer Design

Conventional design for wireless networking only targets performance optimization within isolated protocol layers, and thus inevitable results in suboptimal solutions, since it essentially ignores the intrinsic interdependency among multiple layers. To meet certain application requirements such as energy and delay constraints, a cross-layer design that supports adaptivity and optimization across multiple layers of protocol stack is necessary [5].

In general, cross-layer design is much more than layer coupling and device cooperation. Essentially, it is an adaptation and global optimization problem where successful approaches should provide design flexibility for different applications while still be able to tailor protocol design to the energy constraints in the network devices. In cross-layer design, a system under study should be decomposed into coupled layers in such a way that it could increase the tractability of system performance with respect to different optimization criteria. The interdependence among layers is characterized by the inter-layer information exchange. The information exchange allows each protocol layer to be designed within an integrated framework and jointly optimized under system constraints. However, with so many parameters in the system, the complexity of global optimization is almost prohibitive. Therefore it is imperative to induce layer adaptation to compensate for variations within each individual layer and only exchange a small amount of information among layers.

In general, it is hard to choose what information to exchange and how that information should be adapted in a global manner to the underlying network constraints. [31] provides insightful answer to this question with a generic cross-layer design methodology for energy constrained mobile networks. In particular, the integrated design methodology first identifies direct interactions (key coupling parameters) among layers. For each individual layer, there are three types of parameters that affect the local performance metric and contribute to the global performance criterion: those that directly affect only the local performance metric of individual layer; those that are controllable and directly affect the performance of multiple layers; those that are uncontrollable and directly affect the performance multiple layers. To reduce the design complexity, we need to fix the second and third types of parameters and optimize the local performance metric only with respect to the first type of parameters. Then a generic model for each individual layer is constructed reflecting the influence of the second and third types of parameters on the global performance criterion. Finally, the global performance criterion is optimized with respect to the parameters of the second type. When carrying out the above three-step procedure, local optimization in step one should be consistent with global optimization in step three, since the purpose of local optimization is to filter out the parameter of the first type so as to reduce the design complexity. In general, it is often difficult to obtain complete analytical expressions for either local or global performance metric, thus, to a large extent, cross-layer design has to rely on simulation-based optimization.

The design methodology proposed in [31] is an interesting and innovative attempt to tackle cross-layer design issue in ad hoc networks. In general, cross-layer design is a highly complicated problem and application-specific. There are still a lot of open questions in the understanding and implementation of cross-layer design philosophy for ad hoc wireless networks.

1.3 Summary

In this chapter, we highlighted the importance and objectives of this research dissertation, namely the design and analysis of energy efficient cross-layer protocols for ad hoc networks. The concept of cross-layer design is an innovative networking philosophy to enable the energy constrained operation of ad hoc networks with low cost devices. In a general overview of challenges in ad hoc networking, it becomes evident that conventional design methodologies ignore the interdependency among multiple layers, thus results in suboptimal solutions. In contrast, cross-layer design exploits interaction among layers to achieve globally optimal solutions to ad hoc networking. Finally the discussion of a generic design methodology reveals some insight on cross-layer protocol design with reasonable complexity.

Chapter 2

A Block-Fading Perspective on Random Access Relay Networks

2.1 Introduction

Given current trends in the advancement of technology, wireless networks of limited utility, scale, and lifetime are possible without much further research. However, in order to engineer useful ad hoc wireless networks with long lifetimes and massive scale deployment required for many applications, new analytical tools and approaches to protocol design that reflect recent perspectives on wireless networking are necessary. In addition, the explicit concern with the energy efficiency of wireless networks has evolved into a primary impetus behind the design and implementation of innovative energy saving technologies for ad hoc networks.

In wireless communication, spatial diversity is one of the major solutions to combat channel fading and conserve energy. Traditionally, spatial diversity is achieved by the use of an antenna array at the receiver and/or transmitter. Recently, relaying has been reconsidered as an efficient strategy for exploiting *distributed* spatial diversity present in ad hoc networks, even when each network device only has a single antenna. Relaying as a channel model is first introduced in [32]. A classic single relay channel is a three-terminal network consisting of a source, a relay, and a destination. The source broadcasts a message to both relay and destination, while the relay forwards the message to the destination. By implementing spatial diversity, relaying is able to significantly improve energy efficiency as well as system throughput in ad hoc wireless networks.

There has already been a growing body of literature on the theory of low-energy ad hoc networks and various relaying schemes[1]. However, this research work tends towards three extremes. At one extreme is research that requires that communications follow a cascade of point-to-point links, and

[1] A nonexhaustive list includes [33, 34, 35, 36, 37, 38, 39, 40, 41, 27, 42].

does not allow information-theoretic relaying or cooperative transmission, and therefore does not fully exploit the distributed spatial diversity benefit inherent in wireless networks. As a result, the corresponding protocols under study are de facto multihop. In particular, Gupta and Kumar found that the average throughput furnished to each source in a multihop network diminished to zero as the number of nodes tends to infinity [35]. The fundamental reason for this constriction is that with a uniform traffic pattern, a typical node must expend so much effort forwarding other source's information that few resources remain to transport its own message. This limitation can be alleviated by exploiting mobility in the network, e.g. by having each source transmit to every passing node in the hopes that one of the passing nodes will eventually come close to the destination [37]. It can also be alleviated in dense wireless sensor networks by exploiting the spatial redundancy of the observations [43].

A second extreme considers networks with very few practical constraints, e.g. [36, 40]. Communications is not limited to conventional point-to-point multihopping: Every node receives at least some energy from every other node's transmission. Due to the lack of constraints, it is implicitly assumed that the nodes are capable of coherent *transmission*, that is, distinct terminals must be able to co-phase their transmissions so that they add coherently at a common receiver thereby achieving a beamforming effect. Furthermore, this line of research typically requires that the relay be able to simultaneously receive and transmit. Although these are not realistic assumptions for low-cost networks, the refined protocols incorporating coherent transmission and relaying indeed yield significantly larger rate regions and higher network throughput. For instance, Gupta and Kumar [36] and Gastpar and Vetterli [40] recently show that the constriction on the throughput of multihop networks does not exist in more sophisticated network protocols with multi-terminal coding and relaying schemes. Their results could be considered as a generalized concept of the relay channel [15] to accommodate relay networks with multiple sources, multiple relays, and arbitrary connectivity.

The third extreme is research that allows relaying under more realistic constraints. Høst Madsen [41] and Khojastepour et al [44] consider "cheaper" relay networks that work in a time-division duplexing (TDD) mode and therefore do not receive and transmit simultaneously. The coherent transmission requirement was relaxed by Laneman et al, first for a single relay channel with diversity combining at the destination [45], and then for a multiple relay channel that uses space-time codes to orthogonalize parallel transmissions from the source and relays [46]. Later, Kramer et al [42] rigorously found the capacity for a noncoherent phase fading channel. A twist on relaying involving two sources that may act as relays for each-other was proposed by Sendonaris et al [47, 48, 49] and termed *user cooperation diversity* or *cooperative diversity* [38]. However, most research on practical relaying schemes has to date has only focused on very simple protocols rather than complete random access protocols for the relaying network. A notable exception is the work of Wieselthier et al [34], which considers node-based multicasting protocols that are designed to exploit the so-called *wireless multicast advantage* rather than relying on conventional link-based multihopping. However, that work did not explicitly consider the critical role of automatic repeat request, nor take the information-theoretic perspective of this chapter.

In summary, most of these research results heavily relied on complicated signaling processing and detection techniques which do not readily lend themselves to low-cost network implementation. For instance, coherent transmission requires complicated channel phase estimation at the transmitter, while distributed space time coding in [46] needs accurate symbol alignment as well as carrier frequency synchronization. To engineer energy efficient embedded networks with low-cost devices, in this chapter, we consider a general class of wireless embedded networks that are decomposed into clusters of several low cost radio devices including a source, a destination, and one or more relays. Since each cluster works cooperatively to convey information from source to destination, it could be modelled as an orthogonal random access relay network in which signaling is over a random phase block interference channel, and transmission from the various nodes are non-coherent.

As noted in [33], a non-coherent input relay network can be modelled as a *block fading* channel [50, 51, 52], which in other contexts is called the *block interference* channel [53] and the *Gaussian collision* channel [54]. Because block fading channels are not ergodic, a Shannon-sense capacity does not exist and thus information outage probability is a more relevant performance metric. References [50, 51, 52] established a methodology for determining the information outage probability for block fading channels, which was extended by Foschini and Gans for multi-input, multi-output (MIMO) channels [55]. Building upon [50], Caire and Tuninetti [21] used the renewal-reward theorem of [56] to find achievable bounds on the throughput of hybrid-ARQ protocols operating over block fading channels. Our system model and analysis could be viewed as a generalization of a block fading channel with hybrid FEC/ARQ, the distinction being that a retransmission does not necessarily need to come from the originating device. The model could also be viewed as a constrained version of the relay network considered by [36, 40], where the practical constraint imposed is that the nodes may not transmit coherently. Finally, it could be further considered as a generalization of the orthogonal relaying work of Laneman et al [45] to more sophisticated multiple relay networks managed by an automatic repeat request (ARQ) protocol. Notice that inter-relay communications is permitted in the system, thus it encompasses conventional multihop routing as a special case whereby each transmission is received by just a single, pre-determined terminal. This research project also extends the work of Wieselthier et al [34] to account for ARQ and study the tradeoff between delay and energy consumption from an information-theoretic perspective.

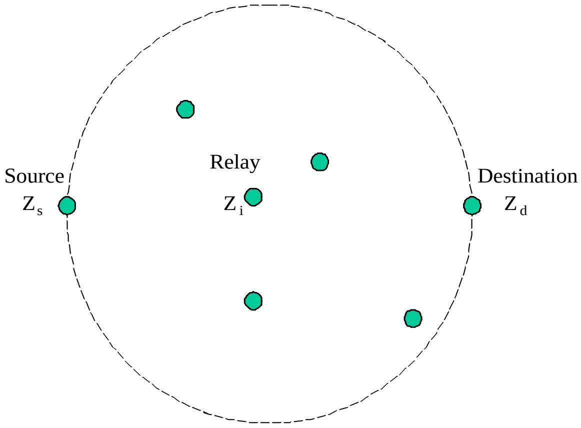

Figure 2.1: A relay network with K nodes.

2.2 System Model

Consider a *cluster* of nodes $\mathcal{N} = \{Z_k : 1 \leq k \leq K\}$ consisting of a *source* $Z_s = Z_1$, a *destination* $Z_d = Z_K$, and $K_r = K - 2$ *relays*. When any node in $\mathcal{N}$ transmits, all nodes also in $\mathcal{N}$ (but not also simultaneously transmitting) may receive the signal over a block fading channel. As we illustrate later, there is a practical upper limit on cluster size. If the cluster has too many nodes, the devices will expend too much of their energy reserves receiving distant transmissions. This limit on size depends mostly on the ratio of the energy consumed while receiving a signal relative to the energy required to transmit it. While small networks (e.g. $K \approx 10$) could consist of just a single cluster (possibly with source, destination, and relays periodically switching roles), larger networks will need to be decomposed into several clusters. Messages that must travel far would be routed from cluster to cluster and a higher level networking protocol will still be needed to handle this routing. However, the networking protocol would only have to route at the cluster-level rather than at the node-level. While this concept is similar to other hierarchical routing protocols like clusterhead gateway switch routing (CGSR) [26], the key difference is that routing within the cluster is now handled implicitly by the retransmission process of the ARQ protocol rather than explicitly by a network-layer routing algorithm.

As we will further discuss in the next chapter, two types of relays are possible: *decoding relays*, which must successfully decode the message before forwarding (*decode-and-forward*), and *amplifying-relays*, which simply repeat an amplified version of the received signal without first decoding (*amplify-and-forward*). More generally, relays may adaptively switch between decoding and amplifying modes. In [38], it is indicated that adaptive decode-and-forward strategies offer the same performance as fixed amplify-and-forward. Therefore, in this chapter we limit our attention to decode-and-forward relaying, which has the side benefit of permitting a more straightforward exposition.

Time is divided into *slots* s, which are commonly assumed to be of equal duration [2]. During slot s, a node may transmit or receive, but not both. If the cluster is part of a chain conveying messages over long distances, then the source (destination) will need to spend roughly half its time acting as the destination (source) of the previous (next) cluster. This could be accomplished through time division duplexing, e.g. a node could act as source for the current cluster during even s and as destination for the previous cluster during odd s.

The source begins by encoding a b bit message into a codeword of length n symbols. The codeword is broken into M blocks/bursts, each of length $L = n/M$ and rate $R = b/L$. The code itself could simply be a *repetition* code, in which case all M blocks are identical and each node will *diversity-combine* [58] all blocks that it has received. More generally, *incremental redundancy* [58] could be used, whereby each block is obtained by puncturing a rate $r_M = R/M$ mother code. With incremental redundancy, a different part of the codeword is transmitted each time, and after the m^{th} block, a receiver will pass the rate $r_m = R/m$ code that it has until then received through its decoder (*code-combining*).

Let $S_m = \{s_1, ...s_m\}$ denote the set of slots over which the first m blocks are sent, where $m \leq M$, K is the maximum transmission blocks allowed per message, also known as the rate constraint. Note that these blocks need not be adjacent. The time $s_m - s_{m-1}$ between transmissions should be chosen to ensure a desired level of temporal decorrelation and could be randomized to mitigate interference (*time-hopping*). The set of nodes that transmit during slot s is denoted $\mathcal{K}(s)$. All transmissions are considered to be *broadcast*, and thus every non-transmitting node in the cluster may receive each transmission. Initially, only the source has knowledge of the codeword, and thus $\mathcal{K}(1) = \{Z_s\}$. During subsequent slots $s, s \geq 2$, *any* node in the cluster that has successfully decoded the message could re-encode it and transmit the next block of the mother code. The exact composition of $\mathcal{K}(s)$ is determined by the protocol being used, which will be discussed in details in later chapters. Under certain random protocols, it is possible that more than one node transmits at the same time. In such a case, each receiver will only use the signal it receives with highest strength (*single-user* detection). The lower strength signals are treated as interference.

Let $\mathbf{x}_k[m] = (x_{k,1}[m], ..., x_{k,L}[m])$ denote the m^{th} block of the codeword as transmitted by node $Z_k \in \mathcal{K}(s_m)$ with average energy per symbol $\mathcal{E}_k[m] = E\{|x_{k,\ell}[m]|^2\}$. Hardware constraints preclude a node from transmitting with symbol energy greater than some maximum value, $\mathcal{E}_{max}$. For the sake of mathematical tractability, we follow [21] and assume circularly symmetric complex Gaussian symbols are transmitted. When simultaneous transmissions occur from two or more

[2] It is sometimes advantageous for the source and relay transmission slots to be of nonidentical length [41, 57], but this leads to a rather inconvenient implementation for relay networks with large rate constraint. A detailed investigation on the effect of nonidentical slot is presented in the next chapter.

nodes, the blocks are identical except for their phases (which are independent and uniform) and their energies (which could be different due to power control). Z_k's transmitted block m is received at $Z_j, j \notin \mathcal{K}(s_m)$, with average energy per symbol $\mathcal{E}_{k,j}[m]$. Signal energy decays exponentially with distance such that $\mathcal{E}_{k,j}[m] = (G_{k,j})^2\mathcal{E}_k[m] = (\lambda_c/4\pi d_o)^2(d_{k,j}/d_o)^{-\mu}\mathcal{E}_k[m]$, where $G_{k,j}$ is the *channel gain* between Z_k and Z_j, $d_{k,j}$ is the distance between Z_k and Z_j, d_o is a reference distance, λ_c is the wavelength of the carrier, and μ is a path loss coefficient with values typically in the range $1 < \mu < 4$ [59]. For notational simplicity, we define average receive SNR $\Gamma_{k,j}[m] = \mathcal{E}_{k,j}[m]/N_o$ and average transmit SNR $\Gamma_k[m] = \mathcal{E}_k[m]/N_o$. Block m is received by Z_j as

$$\mathbf{y}_j[m] \quad = \sum_{k\in\mathcal{K}(s_m)} G_{k,j}c_{k,j}[m]\mathbf{x}_k[m] + \boldsymbol{\nu}_j[m], \tag{2.1}$$

where $\boldsymbol{\nu}_j[m]$ is a vector of circularly symmetric complex Gaussian noise with independent identical distributed (i.i.d.) components with variance N_o, and $c_{k,j}[m]$ is a unit-power complex fading coefficient that describes the random amplitude and phase fluctuations in the channel between nodes k and j (possibly including the effects of shadowing). We assume that the fading coefficient is constant for the duration of a block and varies from block to block. While the fading coefficients may have any arbitrary distribution and correlation (both temporally and spatially), it is common to assume that the coefficients are Rayleigh (or Rician) distributed and independent from block-to-block [51]. In a multi-source network, transmissions not associated with the cluster could be included as either additional terms in the summation or additional Gaussian noise, depending on the nature of the interference. Note that for Gaussian input symbols, both intra- and inter-cluster interference will also be Gaussian, and thus the channel is conditionally Gaussian for the duration of each block even when multiple nodes transmit. In practice, lack of synchronism, especially among the nodes in disjoint clusters, makes this model imperfect.

Because the phase of the channel inputs are independent, transmission is noncoherent (nodes may not co-phase in an effort to achieve a beamforming effect). Since all blocks transmitted within the cluster during slot s_m are equivalent, a reasonable strategy is for each single-user receiver to simply decode the highest strength signal (If the system were to use, for instance, noncoherently detected FSK, then the capture effect of the receiver could be used to implicitly select the highest strength signal.). Specifically, node Z_j will decode the block transmitted by $Z_k : k = \arg\max_i |c_{i,j}[m]|^2\mathcal{E}_{i,j}[m]$ with its *instantaneous* SINR

$$\gamma_j[m] = \frac{\mathcal{E}_{k,j}[m]|c_{k,j}[m]|^2}{N_o + \sum_{i\neq k}\mathcal{E}_{i,j}[m]|c_{i,j}[m]|^2}. \tag{2.2}$$

Due to fading, power control, and random node activity, the instantaneous SINR varies from block to block.

Notice that under this model, an alternative strategy is to transmit identical $\mathbf{x}_k[m] = \mathbf{x}[m], \forall k$, in which case (2.1) can be expressed as

$$\mathbf{y}_j[m] = \mathbf{x}[m] \sum_{k \in \mathcal{K}(s_m)} G_{k,j} c_{k,j}[m] + \boldsymbol{\nu}_j[m] \tag{2.3}$$

and thus there is essentially no intra-cell interference. Rather the channel coefficients (both gain and fading) for the different blocks will add together. Accordingly, the corresponding SNR becomes

$$\gamma_j[m] = \frac{\mathcal{E}_k[m]}{N_o} \left| \sum_{k \in \mathcal{K}(s_m)} G_{k,j} c_{k,j}[m] \right|^2 . \tag{2.4}$$

However, this approach requires that the signals be transmitted in a time-synchronous manner and that the oscillators of the nodes be set to exactly the same frequency. These requirements do not lend themselves to low-cost implementation.

Let $I(\gamma)$ denote the mutual information between the input and output of the channel with instantaneous SINR γ. For Gaussian noise and inputs (and hence, Gaussian interference), $I(\gamma) = \frac{1}{2}\log_2(1+\gamma)$. Note that since γ is random, so is $I(\gamma)$ and therefore a Shannon-sense (ergodic) capacity does not exist [50]. Let $I_j[m]$ denote the mutual information accumulated by node j during the first m transmissions. Under code-combining, the system behaves like a set of m parallel Gaussian channels and thus $I_j[m] = \sum_m I(\gamma_j[m])$ [21]. Alternatively, under diversity-combining the system is a single Gaussian channel with total SINR equal to the sum of the individual SINRs, i.e. $I_j[m] = I(\sum_m \gamma_j[m])$ [21]. Since $\sum_m \log_2(1+\gamma_j[m]) \geq \log_2(1+\sum_m \gamma_j[m])$, code-combining is always at least as good as diversity-combining and is therefore the focus of the remainder of this discussion.

Node Z_j is said to be in an *outage* after the m^{th} block if $I_j[m] < R$. The *outage probability*[3] is then $P_j[m] = Prob\{I_j[m] < R\}$ and can be found by integrating the joint pdf of the m-block channel $p(\gamma_j[1], ..., \gamma_j[m])$ over the *outage region* $\{\gamma_j[1], ..., \gamma_j[m] : I_j[m] < R\}$. We define the *end-to-end outage probability* P_o to be the outage probability at the destination after either all M blocks have been transmitted or a delay constraint of D slots has been reached, whichever comes first.

In a *direct-transmission* system, $K = 2$, and since there is no relay, only the source transmits, $\mathcal{K}(s_m) = \{Z_s\}, \forall m$. When $K > 2$, several relaying strategies are possible. With conventional *multihop*, messages must flow through the cluster as a series of direct transmissions determined *a priori* by a routing algorithm [60]. The destination may not decode the source's direct transmission, even if the instantaneous source-destination SINR is sufficiently high to do so.

If we allow the destination to also "hear" the source, then several other options are possible. First consider a simple system with $K = 3$ and $M = 2$ (but no ARQ). While the first block is

[3] This is also termed *information outage probability* [51] and *outage event probability* [38] and is related to the *outage capacity* [55].

always transmitted by the source, $\mathcal{K}(s_1) = \{Z_s\}$, the second block could again be transmitted by the source or it could instead be transmitted by the relay $Z_r = Z_2$ provided that it decoded the first block, i.e. if $I_r[1] = I(\gamma_{s,r}[1]) > R$. If the relay is in an outage ($I_r[1] < R$), then the transmission ceases after the first block and an end-to-end outage occurs if the source-destination link was in an outage ($I_d[1] = I(\gamma_{s,d}[1]) < R$). Otherwise, the relay will transmit and an end-to-end outage occurs if the parallel channels from source and relay to destination are in an outage [45, 61], $I_d[2] = I(\gamma_{s,d}[1]) + I(\gamma_{r,d}[2]) > R$.

As discussed in the next chapter, a modest amount of adaptability can be introduced by using channel state information (CSI) to guide which of the two nodes transmits the second block [62, 63]. In particular, if the source knows that the relay was in an outage during the first block, then it could transmit the second block instead. Furthermore, if the source and relay know that the relay-destination SNR is less than the source-destination SNR (i.e. $\gamma_{r,d}[2] < \gamma_{s,d}[2])$), then the source could transmit the second block, even if the source-relay link was not in an outage. The detailed analysis of a relay channel with $K = 3$ and $M = 2$ along with its CSI based adaptive protocols could be found in the next chapter. Notice that although these adaptive techniques could be extended to permit multiple relays ($K > 3$) and more transmitted blocks ($M > 2$), the need for each node to have *a priori* knowledge of the CSI of various channels and for the cluster to coordinate transmissions quickly makes this approach unwieldy. The solution that we advocate for selecting which node in a multiple relay network transmits a particular block is to embedded the selection process into the ARQ protocol which will be further studied in chapter 4.

Chapter 3

Rate Constrained Orthogonal Relay Channels

3.1 Introduction

Based on the system model described in chapter 2, we considered a relatively simple relay network with rate constraint $M = 2$ in this chapter. Notice that this network is a constrained version of the more general relay network addressed by Gastpar and Vetterli in [64]. More specifically, the first additional constraint imposed is that transmissions from different nodes in the network must be orthogonal. Although we have assumed that this orthogonality is obtained through time-division multiplexing, it could also be obtained through frequency-division multiplexing provided that the relay maintains causality. We have also imposed a second constraint that each codeword is only broken into two slots, which eases the exposition and permits closed-form expressions for many cases of interest. Although this work could be generalized to situations where the codeword is broken into multiple slots, such an analysis will also need to carefully consider inter-relay communications. In the remainder of our discussion, we will refer to the single-relay network under these constraints as the *constrained single-relay channel* and multiple-relay networks under these constraints as the *constrained multiple-relay channel.* In particular, we use Time Division Multiplexing (TDM) to achieve orthogonal transmissions in the relay channel where the source and relay transmit in orthogonal time slots.

The remainder of this chapter is organized as follows. In section 3.2, we revisit [65] from a slightly different perspective and derive closed-form performance bounds for nonadaptive transmission through the constrained single-relay channel in the presence of block fading. In Section 3.3.2, we propose several new adaptive protocols and find the corresponding capacity and outage probability of each protocol. Like [65], we initially limit the network to no more than two transmissions per message. The first transmission always comes from the source, while the second transmission may again come from the source or alternatively, may come from any of the relays. This limitation is consistent with the use of rate-compatible codes which require that the source creates a low-rate code and then through puncturing, divides this low-rate code into a discrete number of blocks (in our case, just two blocks per codeword) [66]. The benefit of limiting the system to two transmissions is that it allows us to find closed form expressions for many cases of interest, and gives us some insight into the performance of incremental-redundancy based networks with more than just two blocks per codeword. Unlike [65], we do not limit the first and second transmission to be of equal duration. In section 3.4, we characterize the information theoretic limit of user cooperative coding [66] from the adaptive relaying perspective, and provide a more general framework of reference independent of specific of channel coding realization. Finally, Section 3.5 investigates performance in the presence of multiple relays, and shows that performance improves as more relays are added to the network even though only one relay may forward the message and the network is limited to two transmissions per message.

3.2 Nonadaptive Single-Relay Protocols

We assume quasi-static Rayleigh fading for the TDM relay channel. With quasi-static fading, the fading coefficient remains constant over a particular packet (or codeword) and changes independently from packet-to-packet. Thus the channel between two terminals could still be considered to be AWGN for the duration of a particular packet, except that the instantaneous channel SNR is determined by the fading coefficient. Correspondingly the sequence of packet SNRs is i.i.d. exponentially distributed, with average received SNR Γ.

3.2.1 Direct Transmission (DT)

The direct transmission scheme is introduced as a performance benchmark for other relaying schemes. Consider a point-to-point link in which no relaying is involved. For a particular packet transmission, the channel is AWGN with SNR γ_{sd}. Thus the instantaneous channel capacity [67] is defined as

$$C^*_{DT} = \frac{1}{2}\log_2(1+\gamma_{sd}). \tag{3.1}$$

in which γ_{sd} is the instantaneous channel SNR with $E\{\gamma_{sd}\} = \Gamma_{sd}$. If a rate R code is used, then the channel will be in an *outage* whenever $C^*_{DT} < R$, where $\{C^*_{DT} < R\}$ is called the *outage event.* The *outage event probability* (OEP) is found by integrating the pdf of γ_{sd} over the outage event region

$$
\begin{aligned}
Pr\left\{C^*_{DT} \le R\right\} &= \int_0^{2^{2R}-1} \frac{1}{\Gamma_{sd}} \exp\left\{\frac{-\gamma_{sd}}{\Gamma_{sd}}\right\} d\gamma_{sd} \\
&= 1 - \exp\left\{\frac{1-2^{2R}}{\Gamma_{sd}}\right\} \qquad (3.2)
\end{aligned}
$$

3.2.2 Decode-Forward Relaying with Diversity-Combining (DF-MRC)

Decode-forward relaying with diversity-combining uses a repetition code. Because the two blocks are identical, $\alpha = 0.5$. The source begins by generating a rate $R/\alpha = 2R$ code during the first slot (recall that R is the overall rate of the repetition code). If the relay can decode this broadcast, then it will re-encode an identical block and transmit this during the second slot. We assume that the relay can perform perfect error detection and therefore will not transmit if it cannot decode. At the end of the second slot, the destination will maximal ratio combine (MRC) the blocks that it has received from the source and (possibly) from the relay.

With the above definition in mind, the instantaneous capacity of decode-forward relaying with diversity combining was found in [65] to be

$$
C^*_{DF-MRC} = \max\left\{\min\left\{\frac{1}{2}C(\gamma_{s,d}+\gamma_{r,d}), \frac{1}{2}C(\gamma_{s,r})\right\}, \frac{1}{2}C(\gamma_{s,d})\right\} \qquad (3.3)
$$

where $C(x) = \frac{1}{2}\log_2(1+x), x \ge 0$ is the capacity of an AWGN channel with a specific SNR x.

An outage occurs whenever $C^*_{DF-MRC} \le R$. The outage event probability (OEP) can then be found by integrating the joint pdf of the three channel SNRs $p(\gamma_{s,d}, \gamma_{r,d}, \gamma_{s,r})$ over the outage event region

$$
\begin{aligned}
\mathcal{S} &= \left\{C^*_{DF-MRC} \le R\right\} \\
&= \left\{\left[(C(\gamma_{s,r}) > 2R) \cap (C(\gamma_{s,d}+\gamma_{r,d}) < 2R)\right] \cup \left[(C(\gamma_{s,r}) < 2R) \cap (C(\gamma_{s,d}) < 2R)\right]\right\} \quad (3.4)
\end{aligned}
$$

Since these three SNRs are independent exponential random variables, the OEP is found to be:

$$
P_o = \int\int\int_{\mathcal{S}} \frac{1}{\Gamma_{s,d}\Gamma_{r,d}\Gamma_{s,r}} \exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}}\right\} \exp\left\{-\frac{\gamma_{r,d}}{\Gamma_{r,d}}\right\} \exp\left\{-\frac{\gamma_{s,r}}{\Gamma_{s,r}}\right\} d\gamma_{s,d} d\gamma_{r,d} d\gamma_{s,r} \quad (3.5)
$$

By integrating over the outage event region defined (3.4), (3.5) can be reduced to

$$
P_o = 1 - \exp\left\{\frac{1-2^{4R}}{\Gamma_{s,d}}\right\} - \exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\} \frac{\Gamma_{r,d}}{\Gamma_{r,d}-\Gamma_{s,d}} \left(\exp\left\{\frac{1-2^{4R}}{\Gamma_{r,d}}\right\} - \exp\left\{\frac{1-2^{4R}}{\Gamma_{s,d}}\right\}\right) \qquad (3.6)
$$

3.2.3 Decode-Forward Relaying with Code-Combining (DF-CC)

When decode-forward operates in a code-combining mode, the source begins by generating a rate $R = b/n$ incremental redundancy code and broadcasting the first block of the codeword, which is of length $n_1 = \lfloor \alpha n \rfloor$ symbols. If the relay can successfully decode the block, then it will re-encode the message. Then during the second slot, the relay will transmit the second block of $n_2 = n - n_1$ symbols to the destination. At the end of the second slot, the destination will have received the first block from the source and, if the relay could decode the first block, the second block from the relay. It will then pass the entire codeword (with erasures in place of the second block if it was not transmitted by the relay) through its decoder.

Under these conditions, the instantaneous capacity (with α constrained to be a constant) of decode-forward relaying with code-combining is

$$C^*_{DF-CC} = \max \left\{ \min \left\{ \alpha C(\gamma_{s,d}) + \overline{\alpha} C(\gamma_{r,d}), \alpha C(\gamma_{s,r}) \right\}, \alpha C(\gamma_{s,d}) \right\}, \tag{3.7}$$

where $\overline{\alpha} = 1 - \alpha$.

In [68], a similar expression was found for distributed space-time coding. However, the result in [68] required that the two slots be of equal duration ($\alpha = 0.5$), which as we will show later, can be suboptimal.

An outage occurs whenever $C^*_{DF-CC} \leq R$. Thus, the OEP can be found by integrating (3.5) over outage event region

$$\begin{aligned} \mathcal{S} &= \left\{ C^*_{DF-CC} \leq R \right\} \\ &= \left\{ \left[\left(C(\gamma_{s,r}) > \frac{R}{\alpha} \right) \cap \left(\alpha C(\gamma_{s,d}) + \overline{\alpha} C(\gamma_{r,d}) < R \right) \right] \cup \left[\left(C(\gamma_{s,r}) < \frac{R}{\alpha} \right) \cap \left(C(\gamma_{s,d}) < \frac{R}{\alpha} \right) \right] \right\} \end{aligned} \tag{3.8}$$

This result can be explained as follows. First consider the term in (3.8) to the right of the union. The relay is in an outage if $C(\gamma_{s,r}) < R/\alpha$. When the relay is in an outage, an end-to-end outage occurs if the source-destination link is also in an outage, $C(\gamma_{s,d}) < R/\alpha$. On the other hand, if the relay is not in an outage, then the destination will receive a transmission from both source and relay. An end-to-end outage will occur if the parallel channels are in an outage (recalling that the capacity of parallel Gaussian channels adds [67]), e.g. $\alpha C(\gamma_{s,d}) + \overline{\alpha} C(\gamma_{r,d}) < R$.

By integrating over the area defined by (3.8), (3.5) is reduced to

$$\begin{aligned} P_o &= \left(1 - \exp \left\{ \frac{1 - 2^{2R/\alpha}}{\Gamma_{s,d}} \right\} \right) \\ &\quad - \exp \left\{ \frac{1 - 2^{2R/\alpha}}{\Gamma_{s,r}} \right\} \int_0^{2^{2R/\alpha}-1} \frac{1}{\Gamma_{s,d}} \exp \left\{ -\frac{\gamma_{s,d}}{\Gamma_{s,d}} - \frac{1}{\Gamma_{r,d}} \left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma_{s,d})^{\alpha/\bar{\alpha}}} - 1 \right) \right\} d\gamma_{s,d} \end{aligned} \tag{3.9}$$

Now consider the asymptotic behavior of both types of decode-and-forward. As $\Gamma_{r,d} \to \infty$, the OEP of diversity-combining (3.6) is reduced to

$$\lim_{\Gamma_{r,d}\to\infty} P_o = \left(1 - \exp\left\{\frac{1-2^{4R}}{\Gamma_{s,d}}\right\}\right)\left(1 - \exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\}\right). \tag{3.10}$$

Note that when $\alpha = 1/2$, the OEP of code-combining (3.9) will also tend to the limit given by (3.10) as $\Gamma_{r,d} \to \infty$. Likewise, when $\alpha = 0.5$, the OEPs of both diversity-combining (3.6) and code-combining (3.9) are reduced to

$$\lim_{\Gamma_{r,d}\to 0} P_o = 1 - \exp\left\{\frac{1-2^{4R}}{\Gamma_{s,d}}\right\} \tag{3.11}$$

There is an intuitive explanation for this behavior. On the one hand, when the relay does not transmit ($\Gamma_{r,d} \to 0$), the system is reduced to a simple point-to-point link from source to destination. On the other hand, when the relay transmits with infinite power ($\Gamma_{r,d} \to \infty$), the system behaves like a fixed antenna array with one transmit and two receive antennas. Therefore, in extreme SNR regimes, code-combining and diversity-combining will result in the same performance under decode-forward relaying.

3.2.4 Amplify-Forward Relaying (AF)

As with decode-forward relaying with diversity-combining, the two blocks in amplify-forward relaying are identical, and thus $\alpha = 0.5$. The source again broadcasts a rate $R/\alpha = 2R$ code during the first slot. Now, however, the relay always transmits during the second slot. Rather than decoding and re-encoding the message, the relay simply transmits an amplified version of what it received. The destination will then MRC combine what it receives from the source with what it received from the relay.

The capacity of amplify-forward relaying simply becomes $\frac{1}{2}C(\gamma_{s,d} + \gamma_{s,r,d})$, where $\gamma_{s,r,d}$ is the SNR of the relayed path. This SNR will depend on the choice of amplifier gain at the relay. If we let $\xi^2 = \gamma_{s,r}N$ denote the strength of the source's signal at the relay, then the strategy that optimizes the SNR of the destination's MRC combiner is for the relay to use gain $G_2 = 1/\xi^2$. However, it is not easy for the relay to separate the signal power from the noise power and thus it is more feasible to use a gain $G_1 = 1/(\xi^2 + N)$.

When the relay uses gain G_1, the capacity of amplify-forward relaying is [65]

$$C^*_{AF1} = \frac{1}{2}C\left(\gamma_{s,d} + \frac{\gamma_{r,d}\gamma_{s,r}}{1+\gamma_{r,d}+\gamma_{s,r}}\right) \tag{3.12}$$

We can upper bound (3.12) by finding the capacity when using gain $G_2 = 1/\xi^2$. The SNR of the relayed path using this gain is $\gamma_{s,r,d} = \gamma_{r,d}\gamma_{s,r}/(\gamma_{r,d} + \gamma_{s,r})$ [69], and thus the capacity of amplify-forward relaying using gain G_2 is

$$C^*_{AF2} = \frac{1}{2} C \left(\gamma_{s,d} + \frac{\gamma_{r,d}\gamma_{s,r}}{\gamma_{r,d} + \gamma_{s,r}} \right) \tag{3.13}$$

To discriminate these two different types of amplify-forward relaying, we refer to the one with G_1 as type I amplify-forward relaying, while refer to the other with G_2 as type II amplify-forward relaying. Note that $C^*_{AF2} \geq C^*_{AF1}$, therefore type II amplify-forward relaying always performs at least as good as type I amplify-forward relaying regardless of the channel distribution.

In [69], Hasna and Alouini derived the pdf of $\gamma_{s,r,d}$ when $\gamma_{s,r}$ and $\gamma_{r,d}$ are independent and exponential. By applying this result, we find the OEP of type II amplify-forward relaying

$$P_o = \int_0^{2^{4R}-1} P(2^{4R} - 1 - \gamma_{s,d}) \frac{1}{\Gamma_{s,d}} \exp\left\{ -\frac{\gamma_{s,d}}{\Gamma_{s,d}} \right\} d\gamma_{s,d} \tag{3.14}$$

where $P(x)$ is defined as

$$P(x) \;\; = \;\; 1 - \frac{2x}{\sqrt{\Gamma_{s,r}\Gamma_{r,d}}} K_1 \left(\frac{2x}{\sqrt{\Gamma_{s,r}\Gamma_{r,d}}} \right) \exp\left\{ -x \left(\frac{1}{\Gamma_{s,r}} + \frac{1}{\Gamma_{r,d}} \right) \right\}. \tag{3.15}$$

where $K_1(\cdot)$ is the first-order modified Bessel function of the second kind.

For both types of amplify-forward relaying, when $\Gamma_{r,d} \to \infty$,

$$\lim_{\gamma_{r,d} \to \infty} P_o \;\; = \;\; \left(1 - \exp\left\{ \frac{1 - 2^{4R}}{\Gamma_{s,d}} \right\} \right) - \frac{\Gamma_{s,r}}{\Gamma_{s,r} - \Gamma_{s,d}} \left(\exp\left\{ \frac{1 - 2^{4R}}{\Gamma_{s,r}} \right\} - \exp\left\{ \frac{1 - 2^{4R}}{\Gamma_{s,,d}} \right\} \right) \tag{3.16}$$

Likewise, when $\Gamma_{r,d} \to 0$, the OEP reduces to (3.11) ($\alpha = \frac{1}{2}$).

Comparing (3.10) with (3.16), it is apparent that amplify-forward relaying performs asymptotically better than decode-forward relaying when $\Gamma_{r,d} \to \infty$. On the other hand, when $\gamma_{s,r} \to \infty$, $C^*_{AF} = \frac{1}{2} C\left(\gamma_{s,d} + \gamma_{r,d}\right)$, which is identical to the capacity of decode-forward with diversity-combining. However as $\gamma_{s,r} \to \infty$, the capacity of decode-forward with code-combining is $C^*_{DF2} = \frac{1}{2}\left(C(\gamma_{s,d}) + C(\gamma_{r,d})\right)$, assuming $\alpha = 1/2$. Thus, when the source-relay link is reliable, decode-forward will outperform amplify-forward, provided that code-combining is used.

3.3 Single-Relay Adaptive Protocols

An adaptive protocol was investigated in [65] and shown to achieve full diversity for the single-relay channel at asymptotically low OEP. However, the protocol proposed in [65] is based upon repetition coding and diversity-combining. As shown in the last section, a system with incremental redundancy and code-combining will achieve better performance than its diversity-combining counterpart, especially at low SNR. Therefore, we propose several new adaptive protocols, among which, three use code-combining while the other one uses diversity-combining.

In the following discussion of the adaptive protocols, we will first postulate the capacity of each of the adaptive protocols. Once the target capacity is given, then we will discuss the operation of the protocol that achieves the targeted capacity. Finally, we investigate the OEP performance of the protocol.

3.3.1 Source Adaptive Protocols

In the previously discussed nonadaptive protocols, only the relay transmits the second block (or it was not transmitted at all). It is never transmitted by the source. However, there are situations when it may be advantageous for the source to transmit the second block instead of the relay. Source adaptive protocols adaptively switch the responsibility of transmitting the second block between the source and relay. Below, we present two pairs of source adaptive protocols, which differ by the amount of channel state information (CSI) used by the protocol and the type of combining performed by the destination. The first pair of protocols require that the source knows $\gamma_{s,r}$ while the second pair of protocols requires that source knows $\gamma_{s,d}, \gamma_{s,r}$, and $\gamma_{r,d}$ (and the source should direct the transmission mode of the relay in the first block).

For each pair of protocols we consider a diversity-combining version and a code-combining version.

Source Adaptive Protocol A with Diversity Combining (SA-MRC)

Source adaptive protocol 1 (SA-MRC) was originally proposed in [65] and has capacity

$$C^*_{SA-MRC} = \begin{cases} \frac{1}{2}C(2\gamma_{s,d}) & \text{if } \gamma_{s,r} \le 2^{4R}-1 \\ \frac{1}{2}C(\gamma_{s,d}+\gamma_{r,d}) & \text{otherwise} \end{cases} \tag{3.17}$$

SA-MRC uses a repetition code and diversity-combining. The decision rule for deciding which node transmits the second block is as follows:

- If $\gamma_{s,r} < 2^{4R}-1$, the source transmits the second block.

- If $\gamma_{s,r} \ge 2^{4R}-1$, the relay transmits the second block.

Thus, the source will transmit the second block if the relay was in an outage.

We derive its corresponding OEP as

$$\begin{aligned} P_o &= \left(1-\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\}\right)\left(1-\exp\left\{\frac{0.5-2^{4R-1}}{\Gamma_{s,d}}\right\}\right)+\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\}\left(1-\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,d}}\right\}\right) \\ &\quad -\frac{\Gamma_{r,d}}{\Gamma_{r,d}-\Gamma_{s,d}}\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\}\left(\exp\left\{\frac{1-2^{4R}}{\Gamma_{r,d}}\right\}-\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,d}}\right\}\right). \end{aligned} \tag{3.18}$$

Source Adaptive Protocol A with Code Combining (SA-CC)

SA-MRC is not optimal because it only uses diversity-combining. Performance could be improved by using the same rule for deciding which node transmits the second block, but instead using incremental redundancy and code-combining. This is the basis of our first new protocol, SA-CC, which has capacity

$$C^*_{SA-CC} = \begin{cases} C(\gamma_{s,d}) & \text{if } \gamma_{s,r} \leq 2^{2R/\alpha} - 1 \\ \alpha C(\gamma_{s,d}) + \bar{\alpha} C(\gamma_{r,d}) & \text{otherwise} \end{cases} \tag{3.19}$$

With SA-CC, the source generates a rate $R = b/n$ code and sends the first $n_1 = \lfloor \alpha n \rfloor$ symbols during the first slot. If the relay was not in an outage, then it can decode the first block, re-encode the codeword, and transmit the second block of $n_2 = n - n_1$ symbols during the second slot. If instead the relay was in an outage, then the source will transmit the second block. Thus, SA-CC uses a decision rule which is very similar to that used by SA-MRC:

- If $\gamma_{s,r} < 2^{2R/\alpha} - 1$, the source transmits the second block.
- If $\gamma_{s,r} \geq 2^{2R/\alpha} - 1$, the relay transmits the second block.

Aside from the inclusion of α into the decision rule, the key difference between SA-MRC and SA-CC is in the type of coding (repetition vs. incremental redundancy) and combining (diversity- vs. code-combining).

Accordingly, the OEP of SA-CC is

$$\begin{aligned} P_o \;=\;& \left(1 - \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\right)\left(1 - \exp\left\{\frac{1-2^{2R}}{\Gamma_{s,d}}\right\}\right) + \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\left(1 - \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,d}}\right\}\right) \\ & - \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\int_0^{2^{2R/\alpha}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}} - \frac{1}{\Gamma_{r,d}}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma_{s,d})^{\alpha/\bar{\alpha}}} - 1\right)\right\} d\gamma_{s,d} \end{aligned} \tag{3.20}$$

Source Adaptive Protocol B with Diversity Combining (SB-MRC)

The second new adaptive protocol we propose uses repetition coding and diversity-combining. It has target capacity

$$\begin{aligned} C^*_{SB-MRC} &= \max\{C^*_{DF-MRC}, C^*_{DTR}\} \\ &= \frac{1}{2}\max\left\{\min\left\{C(\gamma_{s,d}+\gamma_{r,d}), C(\gamma_{s,r})\right\}, C(2\gamma_{s,d})\right\} \end{aligned}$$

where $C^*_{DTR} = \frac{1}{2}C(2\gamma_{s,d})$ is the instantaneous channel capacity of direct transmission from source-destination (no relaying) with repetition coding. Since, the source transmits the same code during both slots, the SNRs add due to diversity-combining.

The corresponding outage event region is

$$\begin{aligned}\mathcal{S} &= \{C^*_{SB-MRC} \leq R\} \\ &= \{[(C(\gamma_{s,r}) < 2R) \cap (C(2\gamma_{s,d}) < 2R)] \cup [(C(\gamma_{s,r}) > 2R) \cap (C(\gamma_{s,d} + \gamma_{r,d}) < 2R) \cap (C(2\gamma_{s,d}) < 2R)]\},\end{aligned} \tag{3.21}$$

Based on (3.21), we find that SB-MRC uses the following decision rule:

- If $\gamma_{s,r} < 2^{4R} - 1$, the source transmits the second block.
- If $\gamma_{s,r} \geq 2^{4R} - 1$,
  - If $\gamma_{r,d} > \gamma_{s,d}$, the relay transmits the second block.
  - Otherwise, the source transmits the second block

In addition, the OEP is found as

$$\begin{aligned}P_o &= \left(1 - \exp\left\{\frac{0.5 - 2^{4R-1}}{\Gamma_{s,d}}\right\}\right) \\ &\quad -\frac{\Gamma_{r,d}}{\Gamma_{r,d} - \Gamma_{s,d}} \exp\left\{\frac{1 - 2^{4R}}{\Gamma_{s,r}}\right\} \left(\exp\left\{\frac{1 - 2^{4R}}{\Gamma_{r,d}}\right\} - \exp\left\{(0.5 - 2^{4R-1})\left(\frac{1}{\Gamma_{s,d}} + \frac{1}{\Gamma_{r,d}}\right)\right\}\right).\end{aligned} \tag{3.22}$$

Basically, SB-MRC is a refinement over SA-MRC whereby the source will sometimes transmit the second block even if the relay is not in an outage. In particular, the source will transmit whenever the channel from relay to destination is inferior to the channel from source to destination.

Source Adaptive Protocol B with Code Combining (SB-CC)

The third new adaptive protocol we propose is an incremental redundancy based version of SB-MRC which has a target capacity:

$$\begin{aligned}C^*_{SB-CC} &= \max\{C^*_{DF-CC}, C^*_{DT}\} \\ &= \max\{\min\{\alpha C(\gamma_{s,d}) + \overline{\alpha} C(\gamma_{r,d}), \alpha C(\gamma_{s,r})\}, C(\gamma_{s,d})\}\end{aligned} \tag{3.23}$$

The corresponding outage event region is

$$\begin{aligned}\mathcal{S} &= \{C^*_{SB-CC} \leq R\} \\ &= \left\{\left[\left(C(\gamma_{s,r}) < \frac{R}{\alpha}\right) \cap (C(\gamma_{s,d}) < R)\right]\right. \\ &\quad \left.\cup \left[\left(C(\gamma_{s,r}) > \frac{R}{\alpha}\right) \cap (\alpha C(\gamma_{s,d}) + \overline{\alpha} C(\gamma_{r,d}) < R) \cap (C(\gamma_{s,d}) < R)\right]\right\}.\end{aligned} \tag{3.24}$$

According to (3.24), we have the following decision rule for SB-CC:

- If $\gamma_{s,r} < 2^{2R/\alpha} - 1$, the source transmits the second block.
- If $\gamma_{s,r} \geq 2^{2R/\alpha} - 1$,
  - If $\gamma_{r,d} > \gamma_{s,d}$, the relay transmits the second block.
  - Otherwise, the source transmits the second block.

Accordingly the OEP of SB-CC is found to be

$$\begin{aligned} P_o &= \left(1 - \exp\left\{\frac{1-2^{2R}}{\Gamma_{s,d}}\right\}\right) \\ &\quad - \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\} \int_0^{2^{2R}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}} - \frac{1}{\Gamma_{r,d}}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma_{s,d})^{\alpha/\bar{\alpha}}} - 1\right)\right\} d\gamma_{s,d} \end{aligned} \tag{3.25}$$

3.3.2 Relay Adaptive Protocol

With the relay adaptive protocol, the relay will switch between decode-forward relaying and amplify-forward relaying based upon the channel statistics. In [65], a relay adaptive protocol based on repetition coding and diversity-combining was proposed. Here, we propose a new relay adaptive protocol based on incremental redundancy. More specifically, the target channel capacity is

$$\begin{aligned} C^*_{RA} &= \max\{C^*_{DF-CC}, C^*_{AF2}\} \\ &= \max\left\{\min\left\{\frac{1}{2}C(\gamma_{s,d}) + \frac{1}{2}C(\gamma_{r,d}), \frac{1}{2}C(\gamma_{s,r})\right\}, \frac{1}{2}C\left(\gamma_{s,d} + \frac{\gamma_{r,d}\gamma_{s,r}}{\gamma_{r,d}+\gamma_{s,r}}\right)\right\}. \end{aligned} \tag{3.26}$$

We use type II amplify-forward relaying for its analytical convenience and set $\alpha = \frac{1}{2}$.

An outage event will occur whenever the channel SNRs are in

$$\begin{aligned} \mathcal{S} &= \{C^*_{RA} \leq R\} \\ &= \left\{\left[(C(\gamma_{s,r}) < 2R) \cap \left(C\left(\gamma_{s,d} + \frac{\gamma_{r,d}\gamma_{s,r}}{\gamma_{r,d}+\gamma_{s,r}}\right) < 2R\right)\right] \cup \right. \\ &\quad \left. [(C(\gamma_{s,r}) > 2R) \cap (C(\gamma_{s,d}) + C(\gamma_{r,d}) < 2R)]\right\} \end{aligned} \tag{3.27}$$

where the last step follows from the fact that

$$C(\gamma_{s,d}) + C(\gamma_{r,d}) \geq C\left(\gamma_{s,d} + \frac{\gamma_{r,d}\gamma_{s,r}}{\gamma_{r,d}+\gamma_{s,r}}\right) \tag{3.28}$$

Based on (3.27), we have the following relay adaptive protocol:

- The source transmits during s_1, while the relay transmits during s_2.
- If $\gamma_{s,r} < 2^{4R} - 1$, the relay will use type II amplify-forward relaying.

- If $\gamma_{s,r} \geq 2^{4R} - 1$, the relay will decode and forward the source message using incremental redundancy.

The protocol is intuitively natural. Since, whenever $\gamma_{s,r} \leq 2^{4R} - 1$, the source-relay link fails, thus the relay doesn't transmit at all in the decode-forward mode. In order to achieve diversity, the relay has to switch to amplify-forward mode. On the other hand if $\gamma_{s,r} \geq 2^{4R} - 1$, the source-relay link becomes reliable. The relay will switch to decode-forward mode due to (3.28).

The OEP is found as:

$$\begin{aligned} P_o \;=\;& \exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\}\left(1-\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,d}}\right\}\right) \\ & -\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\}\int_0^{2^{4R}-1}\frac{1}{\Gamma_{s,d}}\exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}}-\frac{1}{\Gamma_{r,d}}\left(\frac{2^{4R}}{(1+\gamma_{s,d})}-1\right)\right\}d\gamma_{s,d} \\ & +\left(1-\exp\left\{\frac{1-2^{4R}}{\Gamma_{s,r}}\right\}\right)Pr\left[\left\{\gamma_{s,d}+\frac{\gamma_{r,d}\gamma_{s,r}}{\gamma_{r,d}+\gamma_{s,r}}<2^{4R}-1\right\} \mid \left\{\gamma_{s,r}<2^{4R}-1\right\}\right] \end{aligned} \tag{3.29}$$

where $Pr\left[y \mid x\right]$ stands for the conditional probability of y given x. A combination of numerical integration and Monte Carlo simulation could be used to calculate (3.29).

3.3.3 Performance of Different Relaying Schemes

So far, we have proposed a variety of new adaptive relaying protocols and derived expressions for their outage probabilities. To visualize their advantage over non-adaptive protocols, we wish to determine the OEP as a function of the channel SNRs. However, visualization is difficult due to the large dimensionality of the problem. For instance, there are three average receive SNRs $(\Gamma_{s,d}, \Gamma_{s,r}, \Gamma_{r,d})$, so it is not feasible to show the OEP as a function of all combinations of SNRs on a single plot. Since the destination never transmits, the performance for a particular topology and channel model can more easily be visualized in terms of the two transmit SNRs, Γ_s and Γ_r. However, a plot of OEP vs. these two SNRs would still be three dimensional and therefore hard to visualize on paper. On way to visualize the OEP performance of different relaying schemes is to compute OEP as a function of transmit E_s/N_o (Γ_s, Γ_r) but choose to just show a slice of this three-dimensional plot for the particular value OEP $= 10^{-2}$. To compute this plot, we assume that the relay is placed halfway between the source and destination, i.e. $Z_s = -5$, $Z_d = 5$, and $Z_r = 0$. A path loss coefficient of $\mu = 3$ and constant $K_o = 10^{-4}$ were used in the channel model. The source transmit SNR Γ_s and relay transmit SNR Γ_r were independently varied and the OEP computed for each (Γ_s, Γ_r) pair and it was noted which pairs resulted in a target source-destination OEP of 10^{-2}.

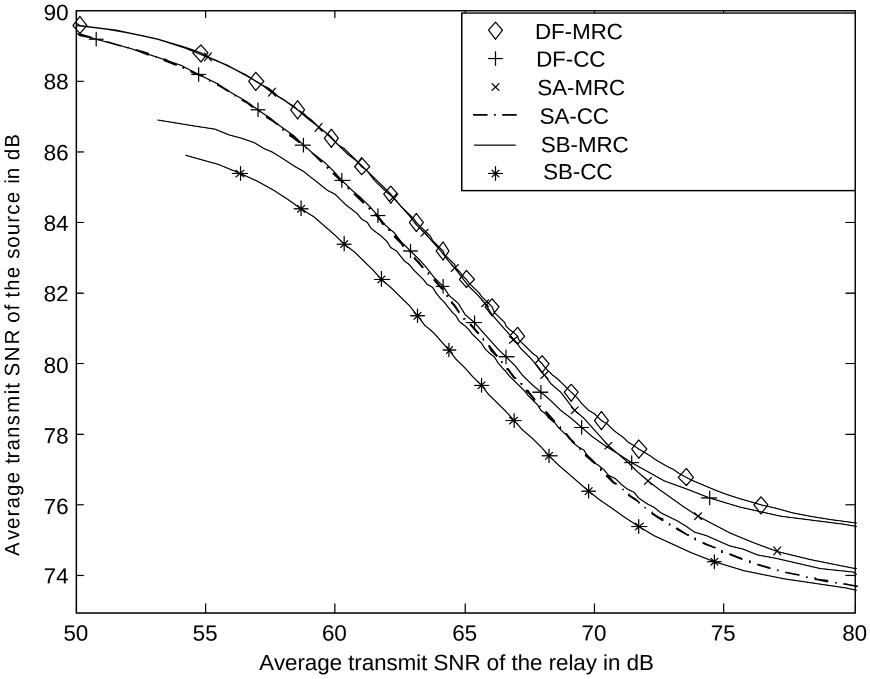

Figure 3.1: Minimum transmit SNR to achieve an end-to-end outage event probability of 10^{-2} with decode-forward schemes as well as several source adaptive protocols in the single-relay channel.

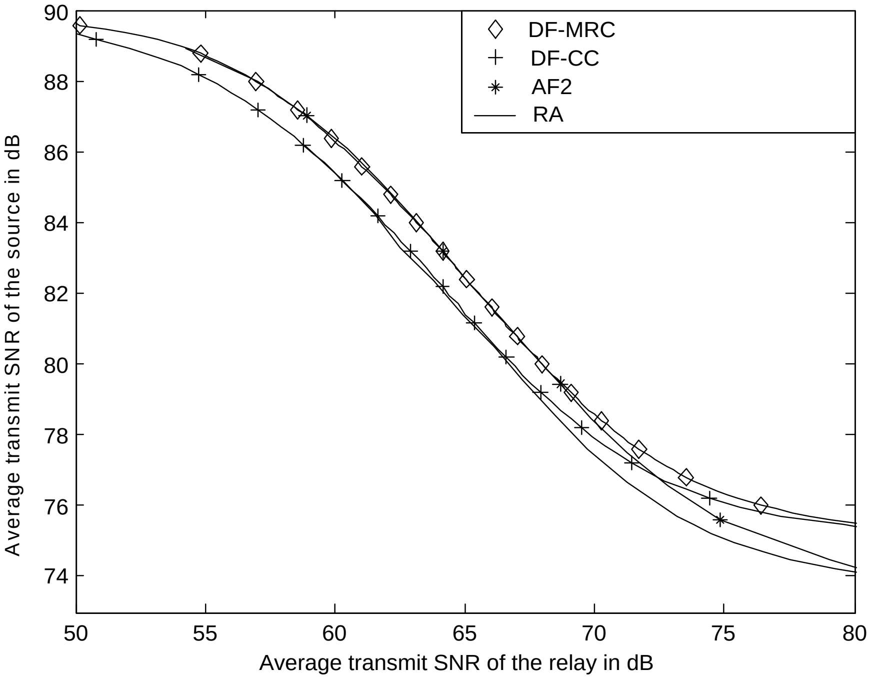

Figure 3.2: Minimum transmit SNR to achieve an end-to-end outage event probability of 10^{-2} with non-adaptive protocols as well as relay adaptive protocol in the single-relay channel.

The contours shown in Fig. 3.1 represent the minimum transmit signal to noise ratios (Γ_s, Γ_r) required to reach an OEP $= 10^{-2}$ for the four source adaptive protocols and the two non-adaptive decode-forward schemes. Similar contours are shown in Fig. 3.2 for the relay adaptive protocol, type-II amplify-forward relaying (AF2), and two non-adaptive decode-forward schemes. In each case, $\alpha = 1/2$.

One observation is that protocols with code-combining are always superior to their diversity-combining based counterpart in two major aspects. First of all, in code-combining capacities add, while in diversity-combining only the SNRs add. Thus code-combining is more energy efficient than diversity-combining. For instance, at an OEP of 10^{-2}, decode-forward with code-combining (DF-CC) is about 1.5 dB more energy efficient than with diversity-combining (DF-MRC). Secondly, with diversity-combining, $\alpha = \frac{1}{2}$, while in code-combining, α could be set as any number between 0 and 1. This gives more flexibility in the system design (the choice of optimal α will be investigated in the next section).

We further observe that adaptive protocols are superior to non-adaptive protocols in the sense that adaptive protocols tend to pick the channel (by choosing the transmission mode) with highest instantaneous capacity. For instance, the relay adaptive protocol is about 1.5 dB more efficient than AF2, although both curves tend to converge at extreme SNR. SB-CC is consistently about 3 dB more efficient than nonadaptive decode-forward, especially when the source is transmitting with high power. In addition, SB-MRC and the RA protocols are asymptotically 1.5 dB more efficient than nonadaptive decode-forward relaying. The contours of different protocols tend to converge under extreme SNR regions. In particular, under high source SNR, SA-MRC and AF2 converge to diversity-combining decode-forward (DF-MRC) relaying, while RA and SA-CC converge to decode-forward relaying with code-combining (DF-CC).

Although adaptive relaying does improve performance, it involves higher complexity in the system design. For instance, to achieve adaptiveness, channel state information (CSI) must be made available to the transmitter a priori. This requires channel feedback. For instance, SA-MRC, SA-CC require the source transmitter to know the instantaneous source/relay SNR $\gamma_{s,r}$. Meanwhile SB-MRC and SB-CC require knowledge of all three instantaneous SNRs. This might appear complicated and even impractical for an actual system. However, this is still feasible under certain circumstances when the source and destination switch roles in a time-division duplexing fashion and the channel coherence time is long enough. If that is the case, SB-MRC and SB-CC could achieve a significant improvement in energy efficiency over both SA-MRC and SA-CC, especially when the source is transmitting with high power.

3.4 User Cooperative Coding: A Relaying Perspective

Relaying has been discovered as an efficient means to collect spatial diversity in both cellular and ad hoc networks. User cooperative coding [66] scheme is a typical example of adaptive relaying. With cooperative coding, users pair up and act as potential relays for each other. Each user encodes blocks of b source bits into n symbol codewords using a prescribed Rate Compatible Punctured Convolutional (RCPC) code [70]. Each n symbol codeword is partitioned into two sets, with the first set being a n_1 symbol punctured codeword, and the second set being the $n_2 = n - n_1$ remaining symbols of the same codeword. Time is divided in two slots. During the first slot, each user broadcasts its first set of n_1 symbols. After the first slot, if a particular user can decode the data from the other, it will calculate and forward the other user's second set of n_2 remaining symbols to the destination during the second slot. Otherwise, the user will send its own n_2 symbols. We notice that the study of user cooperative coding has been primarily focusing on some specific Forward Error Correction codes such as Rate Compatible Punctured Convolutional codes and Rate Compatible Punctured Turbo codes. In this section, we will visualize user cooperative coding from an adaptive relaying perspective. Through investigating its information theoretic limit we are able to provide a general framework of reference for the system design independent of any particular coding schemes.

Notice that the RCPC code based cooperative coding [66] is closely related to the source adaptive relaying protocols. In fact, under the assumption of symmetric inter-user channels, it could be considered as a special case of the source adaptive protocol SA-CC. After all, user cooperative coding is essentially a twist on the adaptive relaying concept for cellular networks where one of the users acts as source while the other as relay. Therefore user cooperative coding should not be restricted to any particular code or any specific protocol. In general, user cooperative coding should be generalized to a much broader concept that could incorporate various adaptive relaying protocols, including source adaptive protocols and relay adaptive protocols. However, in a cellular environment, the source adaptive protocols are more appropriate for user cooperative coding, since the relay adaptive protocol involves amplify-forward scheme that is uncommon in cellular network. In addition, primary results in the last section indicate that protocols under incremental redundancy and code combining always outperforms their diversity combining based counterpart, therefore we will only focus on code combining based source adaptive protocols, e.g. source adaptive protocol SA-CC and source adaptive protocol SB-CC.

In general, we are interested in evaluating cooperative coding from the following three aspects:

- When should we use cooperative coding, e.g. under what circumstance cooperation performs better than direct transmission?

- What is the optimal user cooperation rate?
- How sensitive is the user OEP to the user cooperation rate?

To answer these questions, we need to establish links between the parameters of adaptive relaying and user cooperative coding. In particular, with cooperative coding, users A and B pair up and act as potential relays for each other. User A acts as the source of its own data, while user B as a potential relay for user A and vice versa. Correspondingly, the average inter-user channel SNRs between cooperative users become $\Gamma_{s,r}$, e.g. the average link SNR between source and relay in adaptive relaying; the average channel SNR between user A and destination becomes $\Gamma_{s,d}$; likewise, the average channel SNR between user B and destination becomes $\Gamma_{r,d}$. Finally, the *user cooperation rate* is defined as $\overline{\alpha} = 1 - \alpha$ in adaptive relaying. Worthy of comment is the fact that inter-user channels between two users are generally different in cellular network, i.e. the channel from user A to user B is different from the channel from user B to user A, since cellular network will assign an independent channel to each individual user (preferably separated by different code sequences, time slots or frequency bands). However to facilitate the analysis, the inter-user channels are assumed to be symmetric. Extension to nonsymmetric inter-user channels is straightforward.

Finally, we need to be aware of the fact that there are actually two users involved in user cooperation, therefore the answers to the three questions should be jointly decided by both users. For instance, sometimes user A might favor user cooperation while user B favors direct transmission. If user A has higher priority, then the system should launch user cooperation and vice versa. In the following subsections, we only provide an analytical tool to answer these questions from a single user perspective. More specifically, the analysis only find out whether user A favors user cooperation or not; what is the best cooperation rate for user A. The similar analysis could be applied for user B. Once analytical results of both users are obtained, joint decisions could be made on whether the system should launch user cooperation or not, and what is the best cooperation rate for the system. The joint decisions depend on different applications and user priority, thus will not be covered in our analysis.

3.4.1 Cooperative Coding under Source Adaptive Protocol SA-CC

As mentioned earlier, the RCPC code based cooperative coding [66] is a special case of the source adaptive protocol SA-CC. Therefore, the performance of each individual user involved in cooperative coding could be predicted by the Outage Event Probability (3.20) of SA-CC, e.g.

$$\begin{aligned}P_{coop} &= \left(1-\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\right)\left(1-\exp\left\{\frac{1-2^{2R}}{\Gamma_{s,d}}\right\}\right)+\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\left(1-\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,d}}\right\}\right)\\ &\quad -\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\int_0^{2^{2R/\alpha}-1}\frac{1}{\Gamma_{s,d}}\exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}}-\frac{1}{\Gamma_{r,d}}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma_{s,d})^{\alpha/\bar{\alpha}}}-1\right)\right\}d\gamma_{s,d}\end{aligned}$$

In addition, the OEP of noncooperative/direct transmission is characterized by

$$P_{non} = 1-\exp\left\{\frac{1-2^{2R}}{\Gamma_{s,d}}\right\}$$

Therefore, a performance metric dedicated to compared cooperative vs. noncooperative transmission schemes could be defined as

$$\begin{aligned}M_{coop} &= P_{coop}-P_{non}\\ &= \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\left(\exp\left\{\frac{1-2^{2R}}{\Gamma_{s,d}}\right\}-\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,d}}\right\}\right)\\ &\quad -\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\int_0^{2^{2R/\alpha}-1}\frac{1}{\Gamma_{s,d}}\exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}}-\frac{1}{\Gamma_{r,d}}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma_{s,d})^{\alpha/\bar{\alpha}}}-1\right)\right\}d\gamma_{s,d}\end{aligned} \tag{3.30}$$

The major purpose is to determine the user cooperation rate set $S_{\bar{\alpha}}$ where user cooperation always performs better than non cooperative transmission, e.g.

$$S_{\bar{\alpha}} = \{\bar{\alpha} : M_{coop} < 0, 0 \leq \bar{\alpha} < 1\} \tag{3.31}$$

Notice that the common term in (3.30) $e^{\left(1-2^{2R/\alpha}\right)/\Gamma_{s,r}}$ is always positive, therefore $S_{\bar{\alpha}}$ only depends on $\Gamma_{s,d}$ and $\Gamma_{r,d}$. More specifically, (3.34) could be further reduced to

$$S_{\bar{\alpha}}(\Gamma_{s,d},\Gamma_{r,d}) = \{\bar{\alpha} : M_{equ}(\bar{\alpha},\Gamma_{s,d},\Gamma_{r,d}) < 0, 0 \leq \bar{\alpha} \leq 1\} \tag{3.32}$$

where

$$\begin{aligned}M_{equ}(\bar{\alpha},\Gamma_{s,d},\Gamma_{r,d}) &= \exp\left\{\frac{1-2^{2R}}{\Gamma_{s,d}}\right\}-\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,d}}\right\}\\ &\quad -\int_0^{2^{2R/\alpha}-1}\frac{1}{\Gamma_{s,d}}\exp\left\{-\frac{\gamma}{\Gamma_{s,d}}-\frac{1}{\Gamma_{r,d}}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma)^{\alpha/\bar{\alpha}}}-1\right)\right\}d\gamma\end{aligned} \tag{3.33}$$

A critical cooperation rate $\bar{\alpha}$ is defined such that

$$\bar{\alpha}_{critical}(\Gamma_{s,d},\Gamma_{r,d}) = \{\bar{\alpha} : M_{equ}(\bar{\alpha},\Gamma_{s,d},\Gamma_{r,d}) = 0, 0 \leq \bar{\alpha} \leq 1\} \tag{3.34}$$

Notice both $S_{\bar{\alpha}}$ and $\bar{\alpha}_{opt}$ are only dependent on user channel SNRs $\Gamma_{s,d}$ and $\Gamma_{r,d}$.

More generally, we could define a three dimensional user cooperation space:

$$\Omega = \{(\bar{\alpha}, \Gamma_{s,d}, \Gamma_{r,d}) : \Gamma_{s,d} > 0, \Gamma_{r,d} > 0, 0 \leq \bar{\alpha} \leq 1\} \tag{3.35}$$

where each user cooperation point has three dimensions, i.e. $\omega = (\bar{\alpha}, \Gamma_{s,d}, \Gamma_{r,d})$ corresponding to a particular user cooperation rate $\bar{\alpha}$ and channel SNR pair $(\Gamma_{s,d}, \Gamma_{r,d})$. Finding $S_{\bar{\alpha}}$ is equivalent to finding a subset $\Lambda_1 \subset \Omega$ such that

$$\Lambda_1 = \{\forall \omega \in \Omega : M_{equ}(\omega) < 0\} \tag{3.36}$$

The corresponding subset $\Lambda_2 \subset \Omega$ is the region such that

$$\Lambda_2 = \{\forall \omega \in \Omega : M_{equ}(\omega) > 0\} \tag{3.37}$$

A *critical user cooperation surface* Ψ is defined to partition Ω into Λ_1 and Λ_2,

$$\Psi = \{\forall \omega \in \Omega : M_{equ}(\omega) = 0\} \tag{3.38}$$

An alternative expression for Ψ is

$$\Psi = \{\forall \omega \in \Omega : \omega = (\bar{\alpha}_{critical}, \Gamma_{s,d}, \Gamma_{r,d})\} \tag{3.39}$$

For example, numerical analysis of (3.32) indicates that when $\Gamma_{s,d} = 40$ dB and $\Gamma_{r,d} = 0$ dB, $S_{\bar{\alpha}}(40, 0) = \{\bar{\alpha} : 0 \leq \bar{\alpha} < 0.63\}$, thus $\bar{\alpha}_{critical} = 0.63$, $\omega = (0.63, 40, 0)$, and $\omega \in \Psi$. An immediate question arises from this counter-intuitive result is why to relay the source message when the source channel $\Gamma_{s,d}$ is much better than the relay channel $\Gamma_{s,d}$. The reason is twofold. On the one hand, this user cooperation scheme is based on adaptive relaying protocol SA-CC where user cooperation is reduced to direct transmission whenever the source-relay link fails. On the other hand, once the source-relay channel is able to reliably convey the source message, user cooperation is able to deliver extra diversity benefit due to additional transmission path in the system.

We applied numerical analysis to find out Ψ according to (3.38) and plotted out this three dimensional surface in Fig. 3.3. However, due to the stability problem in the numerical analysis, we only search for a subset $\bar{\Omega} \subset \Omega$ for Ψ,where

$$\bar{\Omega} = \{\forall \quad \omega = (\bar{\alpha}, \Gamma_{s,d}, \Gamma_{r,d})) : 0.05 \leq \bar{\alpha} \leq 0.95\} \tag{3.40}$$

Therefore, $\bar{\alpha}_{critical}$ is always upper-bounded by 0.95 and lower-bounded by 0.05. With this restriction on the surface $\bar{\Psi}$, what we have found and plotted in Fig. 3.3 is a distorted version of Ψ. In particular,

$$\bar{\Psi} = \{\forall \quad \omega \in \Omega : \omega = (\bar{\alpha}, \Gamma_{s,d}, \Gamma_{r,d}), \bar{\alpha} = \min\{0.95, \max\{0.05, \bar{\alpha}_{critical}\}\}\} \tag{3.41}$$

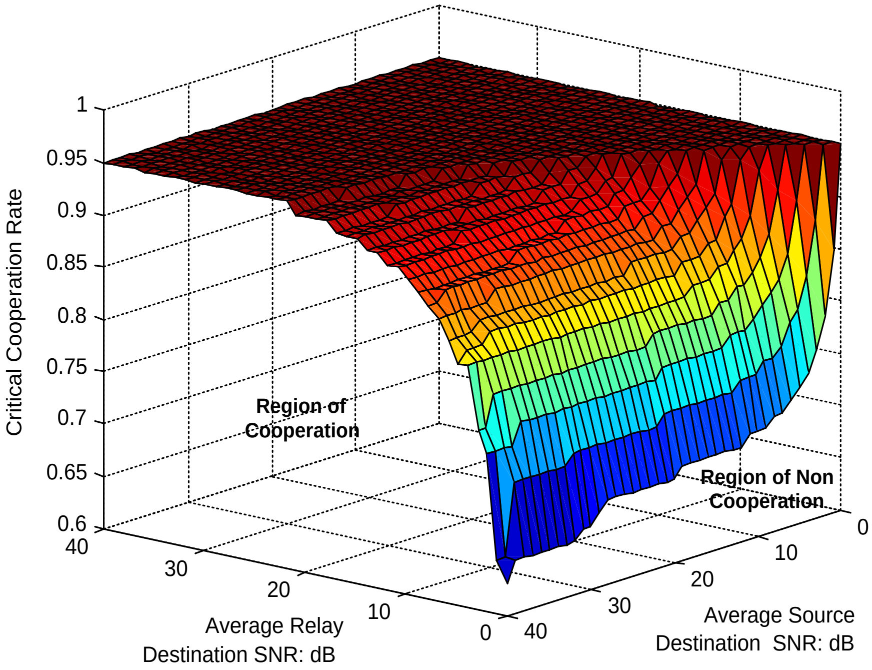

Figure 3.3: The critical user cooperation surface Ψ splits the user cooperation space Ω into two disjoint subsets Λ_1 and Λ_2, where Λ_1 favors user cooperation and Λ_2 favors direct transmission.

Thus any points on the surface $\bar{\Psi}$ close to or on the planes of $\bar{\alpha} = 0.05$ and $\bar{\alpha} = 0.95$ are only an approximation of actual $\bar{\alpha}_{critical}$. However, from a practical design perspective, users with $\bar{\alpha}_{critical}$ upper-bounded by 0.95 in the user cooperation space are most encouraged to apply user cooperation, since almost any cooperation rate is preferable over direct transmission. On the other hand, users with $\bar{\alpha}_{critical}$ lower-bounded by 0.05 are discouraged to apply user cooperation, since almost all cooperation rates will result in performance deterioration from noncooperative transmission. As mentioned earlier, $\bar{\Psi}$ splits user cooperation space into two regions, where the region above $\bar{\Psi}$ corresponds to Λ_1 which favors user cooperation, while the region below $\bar{\Psi}$ corresponds to Λ_2 which favors direct transmission. Worthy of comment is the fact that the channel condition between source and relay does not affect the choice between direct transmission and user cooperation as long as the inter-user channels are assumed to be symmetric. We further notice that as long as $\Gamma_{r,d} \geq \Gamma_{s,d}$, it is always worthwhile to apply user cooperation.

To find out the optimal user cooperation rate, we define

$$\bar{\alpha}_{opt} = \underset{0 \leq \bar{\alpha} \leq 1}{\operatorname{argmin}} P_{coop} \tag{3.42}$$

Since $P_{coop} = P_{non} + M_{coop}$ and P_{non} is constant over all $\bar{\alpha}$, minimizing P_{coop} over $\bar{\alpha}$ is equivalent to minimizing M_{coop}. In general, $\bar{\alpha}_{opt}$ is a function of $\Gamma_{s,r}$,$\Gamma_{s,d}$,$\Gamma_{r,d}$, which is hard to visualize in a three dimensional plot. In cellular systems, it is not unrealistic to assume that $\Gamma_{r,d} \geq \Gamma_{s,d} = \Gamma$ due to power control. Therefore, based on this assumption, $\bar{\alpha}_{opt}$ only depends on Γ and $\Gamma_{s,r}$.

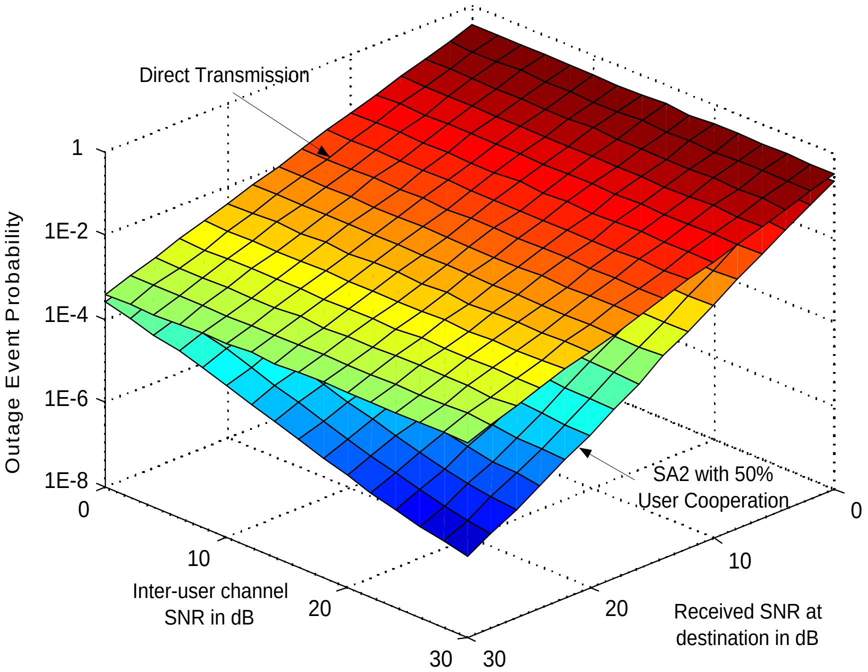

Figure 3.4: The OEP performance comparison of direct transmission scheme vs. source adaptive protocol SA-CC with 50% user cooperation rate.

To have a general idea of how user cooperation could improve performance, we plot in Fig. 3.4 the OEP performance comparison of direct transmission vs. 50% user cooperation under source adaptive protocol SA-CC. We observe that user cooperation schemes significantly outperform direct transmission especially when both average inter-channel SNR Γ_{sr} and the received SNR Γ at the destination are high enough. As inter-channel SNR Γ_{sr} increases, the source-relay link becomes reliable, where user cooperation achieves a diversity order of 2 in block fading channel, while direct transmission only has a diversity order of 1. Fig. 3.5 illustrates the optimal user cooperation rate $\bar{\alpha}$ for each individual SNR pair (Γ_{sr}, Γ) according to (3.42). In Fig. 3.5, we observe that whenever the inter-user channel is reliable, the optimal user cooperation rate is around 50%. On the other hand, when inter-user channel is not reliable but the signal quality at the destination is good, less cooperation leads to much better performance.

In order to investigate the sensitivity of user performance under different user cooperation rate, we compare the OEP of optimal user cooperation against that of 50% user cooperation. In particular, we define optimal user cooperation gain:

$$G_{opt} = 1 - \frac{OEP(\bar{\alpha}_{opt}, \Gamma_{sr}, \Gamma)}{OEP(50\%, \Gamma_{sr}, \Gamma)} \tag{3.43}$$

where $OEP(\bar{\alpha}_{opt}, \Gamma_{sr}, \Gamma)$ denotes the user outage probability under optimal cooperation rate $\bar{\alpha}_{opt}$ at a specific channel SNR pair (Γ_{sr}, Γ), while $OEP(50\%, \Gamma_{sr}, \Gamma)$ denotes the user outage probability under 50% cooperation at the same SNR pair (Γ_{sr}, Γ).

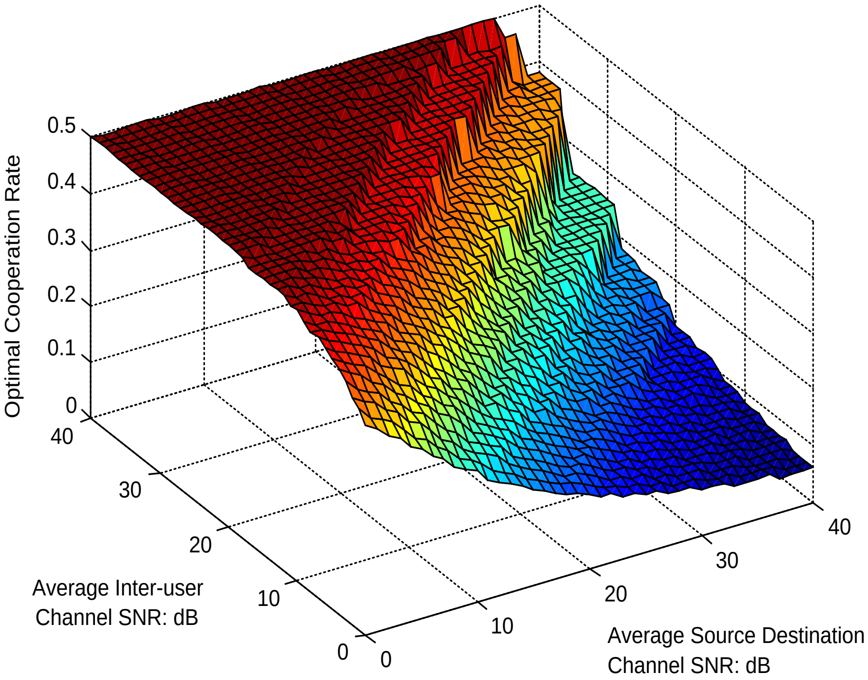

Figure 3.5: The optimal user cooperation rate $\bar{\alpha}$ as a function of $\Gamma_{s,r}$ and Γ under source adaptive protocol SA-CC.

In Fig. 3.6 the optimal user cooperation gain is a function of channel SNR pair (Γ_{sr}, Γ). In general, we observe that user OEP is rather insensitive to user cooperation rate, since the maximum gain is less that 50%. In the area of large Γ_{sr} and small Γ, performance improvement due to optimal cooperation rate over 50% user cooperation is almost negligible. On the other hand, as inter-user channel become less reliable, it is still worthwhile to pick the optimal cooperation rate, because about 50% of performance improvement is achievable.

Finally, when $\Gamma_{s,r} \to \infty$, (3.20) is reduced to

$$P_o = \left(1 - \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma}\right\}\right) - \int_0^{2^{2R/\alpha}-1} \frac{1}{\Gamma} \exp\left\{-\frac{1}{\Gamma}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma)^{\alpha/\bar{\alpha}}} - 1 + \gamma\right)\right\} d\gamma \quad (3.44)$$

This is equivalent to the OEP of direct transmission in over a multiple block fading channel [51]. When the number of blocks is two (as it is here), $\alpha = 1/2$ has the best performance.

3.4.2 Cooperative Coding under Source Adaptive Protocol SB-CC

In the previous, we discussed user cooperation based on source adaptive protocol SA-CC. In this section, we will study a different user cooperation scheme using source adaptive protocol SB-CC. Notice that SB-CC requests each user to have its own CSI as well as its partner user's CSI a priori. User cooperation is only possible when inter-user channel is reliable and the partner user's channel is stronger than the user's own channel. Through our previous discussion on adaptive protocols,

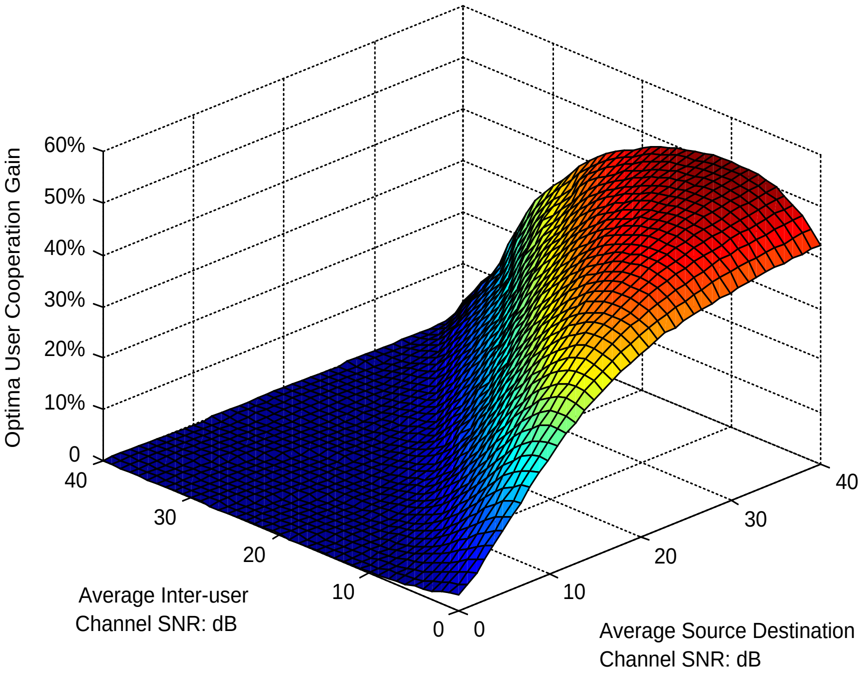

Figure 3.6: The performance improvement of source adaptive protocol SA-CC under optimal cooperation over 50% user cooperation.

we have already reached the conclusion that adaptive protocol SB-CC always outperforms SA-CC. Thus, in this section we will only study its optimal user cooperation rate and its optimal cooperation gain, following a process similar to that of SA-CC. In particular, the performance of each individual user could be expressed by the Outage Event Probability of SB-CC, e.g.

$$\begin{aligned} P_{coop} &= \left(1 - \exp\left\{\frac{1-2^{2R}}{\Gamma_{s,d}}\right\}\right) \\ &\quad - \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\} \int_0^{2^{2R}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}} - \frac{1}{\Gamma_{r,d}}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma_{s,d})^{\alpha/\bar{\alpha}}} - 1\right)\right\} d\gamma_{s,d} \end{aligned}$$

Likewise,

$$\begin{aligned} M_{coop} &= P_{coop} - P_{non} \\ &= -\exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\} \int_0^{2^{2R}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma_{s,d}}{\Gamma_{s,d}} - \frac{1}{\Gamma_{r,d}}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma_{s,d})^{\alpha/\bar{\alpha}}} - 1\right)\right\} d\gamma_{s,d} \end{aligned} \tag{3.45}$$

Notice that $M_{coop} \leq 0$, thus $S_{\bar{\alpha}} = \{\alpha, 0 < \alpha < 1\}$ (when $\bar{\alpha} = 0$ or $\bar{\alpha} = 1$, SB-CC reduce to direct transmission). Therefore,

$$\Lambda_1 = \{\forall \omega \in \Omega : 0 < \bar{\alpha} < 1\} \tag{3.46}$$

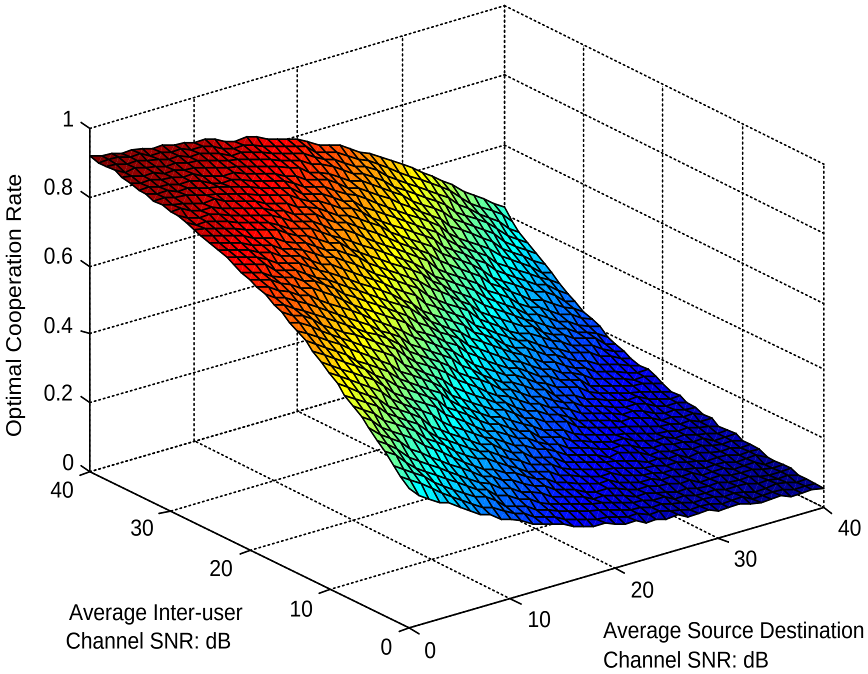

Figure 3.7: The optimal user cooperation rate $\bar{\alpha}$ as a function of $\Gamma_{s,r}$ and Γ under source adaptive protocol SB-CC.

$$\Lambda_2 = \{\phi\} \tag{3.47}$$

Assuming perfect power control, the optimal user cooperation rate is defined as

$$\bar{\alpha}_{opt}(\Gamma_{s,r}, \Gamma) = \underset{0\le\bar{\alpha}\le1}{argmin} - \exp\left\{\frac{1-2^{2R/\alpha}}{\Gamma_{s,r}}\right\}\int_0^{2^{2R}-1}\frac{1}{\Gamma}\exp\left\{-\frac{\gamma_{s,d}}{\Gamma}-\frac{1}{\Gamma}\left(\frac{2^{2R/\bar{\alpha}}}{(1+\gamma)^{\alpha/\bar{\alpha}}}-1\right)\right\}d\gamma \tag{3.48}$$

Under perfect power control, Fig. 3.7 illustrates the optimal user cooperation rate $\bar{\alpha}$ for each individual SNR pair (Γ_{sr}, Γ) according to (3.48). We observe that whenever the inter-user channel is much stronger than the user channel e.g. $\Gamma_{s,r} >> \Gamma = \Gamma_{s,d} = \Gamma_{r,d}$, more user cooperation results in better performance and vice versa. In Fig. 3.8, we observe that whenever the inter-user channel condition is close to that of user channel, e.g. $\Gamma_{s,r} \approx \Gamma$, optimal user cooperation gain is negligible, e.g 50% user cooperation is almost the best choice. While in other two extreme SNR regions where either $\Gamma_{s,r} >> \Gamma$ or $\Gamma_{s,r} << \Gamma$, it is still worthwhile to find out the optimal cooperation rate, since optimal cooperation could achieve up to 50% of performance improvement.

3.5 Multiple-Relay Protocols

Now let's turn our attention to the multiple-relay network, i.e. $K_r > 1$. We assume that the network uses decode-forward relaying and code-combining. We again constrain the network to

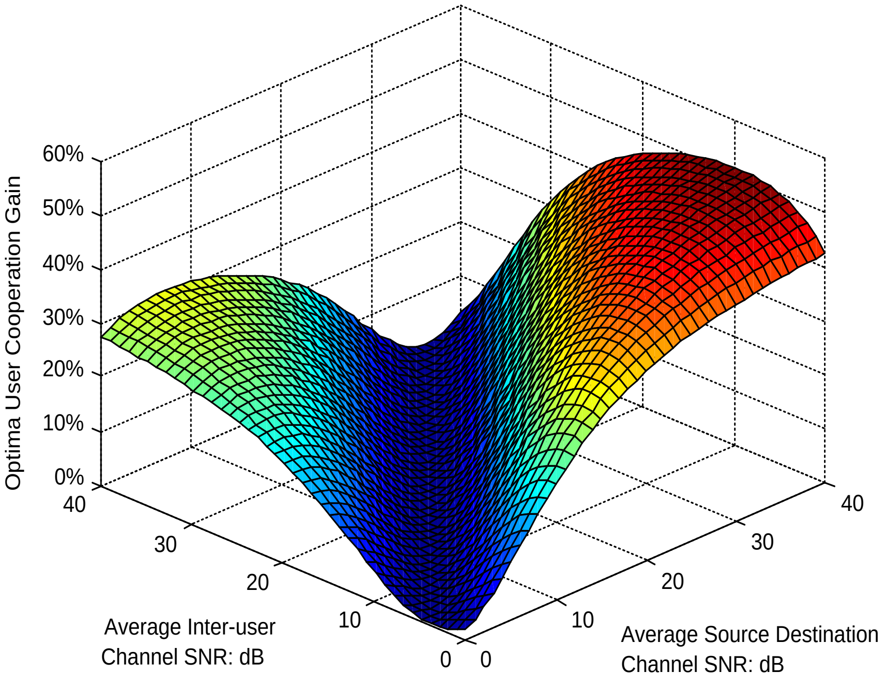

Figure 3.8: The performance improvement of source adaptive protocol SB-CC under optimal cooperation over 50% user cooperation.

two transmissions, e.g. $M = 2$, but now the second block could be transmitted by any relay in $D(s)$ (later we consider an adaptive strategy that also permits the source to transmit the second block). The extension to multiple transmissions is rather complicated in the sense that it involves inter-relay communication.

For analytical convenience, we consider a clustering topology where all the relays are clustered within a small circle. The radius of the circle is much smaller than the source/destination separation. However, within the cluster, the relays are also separated far enough apart such that the channels from the source to each relay are independent; likewise, the channels from each relay to the destination are independent.

Based on the network topology, we have the following SNR constraints:

- $\{\gamma_{s,i}\}$ are i.i.d. exponential random variables with mean $\Gamma_{s,i} \approx \Gamma_{s,r}$, $2 \leq i \leq K-1$.
- $\{\gamma_{i,d}\}$ are i.i.d. exponential random variables with mean $\Gamma_{i,d} \approx \Gamma_{r,d}$, $2 \leq i \leq K-1$.
- The transmit SNRs of the relays $\Gamma_i \approx \Gamma_r$, $2 \leq i \leq K-1$.

To find the OEP of the constrained multiple-relay channel, we first recognize that since the source broadcasts a rate R/α code, the probability that $|D(s)| = k+1$ is

$$Pr\left\{|D(s)| = k+1\right\} = \left(\begin{array}{c} K_r \\ k \end{array}\right) \exp\left\{k\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\left(1-\exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r-k}, 0 \leq k \leq K_r. \tag{3.49}$$

Notice that the source is always in $D(s)$, therefore k denotes the number of relays in the decoding set.

3.5.1 Relay Selection Strategies

Ideally, we assume that a genie will help select the best relay within the decoding set $D(s)$ to forward the message. In practice, without global coordination, simultaneous transmissions from multiple relays might occur which would create multiple-access interference (MAI) (recall that coherent transmissions are not permitted under our constraints). Although the genie-aided performance limits derived in this section work as lower bounds on practical, non-genie aided systems, they reveal some insight on the design and performance of protocols for the constrained multiple-relay channel.

In the following, we consider two different strategies to select the relay node to transmit the second block, namely the optimal selection strategy and random selection strategy. With optimal selection strategy the genie will pick the relay node $Z_i \in D(s)$ with the highest instantaneous SNR $\gamma_{i,d}$ to the destination, while with random selection strategy the genie will randomly pick a relay node $Z_i \in D(s)$, where $2 \leq i \leq K-1$.

More specifically, with optima selection strategy, the instantaneous channel SNR $\gamma_{r,d}$ of the selected relay has a conditional CDF of

$$P(\gamma_{r,d}|\left\{|D(s)| = k+1\right\}) = \left(1-\exp\left\{\frac{-\gamma_{r,d}}{\Gamma_{r,d}}\right\}\right)^{k}, \quad 1 \leq k \leq K_r \tag{3.50}$$

Thus, its corresponding pdf is

$$p(\gamma_{r,d}|\left\{|D(s)| = k+1\right\}) = \frac{k}{\Gamma_{r,d}}\exp\left\{\frac{-\gamma_{r,d}}{\Gamma_{r,d}}\right\}\left(1-\exp\left\{\frac{-\gamma_{r,d}}{\Gamma_{r,d}}\right\}\right)^{k-1}, 1 \leq k \leq K_r \tag{3.51}$$

On the other hand, with random selection strategy, the instantaneous channel SNR $\gamma_{r,d}$ of the selected relay has a conditional pdf

$$p(\gamma_{r,d}|\left\{|D(s)| = k+1\right\}) = \frac{1}{\Gamma_{r,d}}\exp\left\{\frac{-\gamma_{r,d}}{\Gamma_{r,d}}\right\}, 1 \leq k \leq K_r. \tag{3.52}$$

3.5.2 Nonadaptive Protocol

In general, the OEP of the constrained multiple-relay channel with nonadaptive decode-forward relaying can then be expressed as:

$$\begin{aligned} P_o &= \sum_{k=1}^{K_r} Pr\left\{|D(s)| = k+1\right\} Pr\left[\left\{\overline{\alpha} C\left(\gamma_{r,d}\right) + \alpha C\left(\gamma_{s,d}\right) \le R\right\} \mid \left\{|D(s)| = k+1\right\}\right] \\ &\quad + Pr\left\{|D(s)| = 1\right\} Pr\left\{\alpha C\left(\gamma_{s,d}\right) \le R\right\} \end{aligned} \tag{3.53}$$

In particular, the conditional probability

$$Pr\left[\mathcal{A} \mid \left\{|D(s)| = k+1\right\}\right] = \int\int_{\mathcal{A}} \frac{1}{\Gamma_{s,d}} \exp\left\{\frac{-\gamma_{s,d}}{\Gamma_{s,d}}\right\} p(\gamma_{r,d}| \left\{|D(s)| = k+1\right\}) d\gamma_{s,d} d\gamma_{r,d} \tag{3.54}$$

where random event $\mathcal{A} = \left\{\overline{\alpha} C\left(\gamma_{r,d}\right) + \alpha C\left(\gamma_{s,d}\right) \le R\right\}$.

Substituting (3.54) and (3.51) into (3.53), we find the OEP of nonadaptive decode-forward relaying with optimal selection strategy

$$\begin{aligned} P_o &= \sum_{k=0}^{K_r} \begin{pmatrix} K_r \\ k \end{pmatrix} \exp\left\{k \frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\} \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r - k} \\ &\qquad \int_0^{4^{R/\alpha}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma}{\Gamma_{s,d}}\right\} \left(1 - \exp\left\{\frac{1}{\Gamma_{r,d}}\left(1 - \frac{4^{R/\overline{\alpha}}}{(1+\gamma)^{\alpha/\overline{\alpha}}}\right)\right\}\right)^k d\gamma \end{aligned} \tag{3.55}$$

Likewise, with random selection strategy, (3.53) becomes:

$$\begin{aligned} P_o &= \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,d}}\right\}\right) \\ &\quad - \left(1 - \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r}\right) \int_0^{4^{R/\alpha}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma}{\Gamma_{s,d}}\right\} \exp\left\{\frac{1}{\Gamma_{r,d}}\left(1 - \frac{4^{R/\overline{\alpha}}}{(1+\gamma)^{\alpha/\overline{\alpha}}}\right)\right\} d\gamma \end{aligned} \tag{3.56}$$

Both strategies will result in the same OEP expressions in extreme relay-destination SNR. In particular,

$$\lim_{\Gamma_{r,d}\to\infty} P_o = \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,d}}\right\}\right) \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r} \tag{3.57}$$

Likewise,

$$\lim_{\Gamma_{r,d}\to 0} P_o = 1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,d}}\right\} \tag{3.58}$$

3.5.3 Adaptive Protocol

Noticing that the performance of the single-relay channel could be improved by incorporating adaptive protocols, we will investigate the potential advantages of applying adaptive protocols in the multiple-relay channel. Take source adaptive protocol SA-CC as a particular example. With SA-CC, if $|D(s)| = 1$, there is no relay in $D(s)$, therefore the source will transmit the second block during s_2, otherwise the second block will be transmitted by the selected relay (with different selection strategies). Thus, the OEP of the multiple-relay channel with SA-CC is found as:

$$\begin{aligned} P_o &= \sum_{k=0}^{K_r} Pr\left\{|D(s)| = k+1\right\} Pr\left[\left\{\overline{\alpha} C\left(\gamma_{r,d}\right) + \alpha C\left(\gamma_{s,d}\right) \leq R\right\} \mid \left\{|D(s)| = k+1\right\}\right] \\ &\quad + Pr\left\{|D(s)| = 1\right\} Pr\left\{C\left(\gamma_{s,d}\right) \leq R\right\} \end{aligned} \tag{3.59}$$

With optimal selection strategy, (3.59) becomes

$$\begin{aligned} P_o &= \sum_{k=0}^{K_r} \begin{pmatrix} K_r \\ k \end{pmatrix} \exp\left\{k \frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\} \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r-k} \\ &\quad \int_0^{4^{R/\alpha}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma}{\Gamma_{s,d}}\right\} \left(1 - \exp\left\{\frac{1}{\Gamma_{r,d}}\left(1 - \frac{4^{R/\overline{\alpha}}}{(1+\gamma)^{\alpha/\overline{\alpha}}}\right)\right\}\right)^k d\gamma \\ &\quad + \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r} \left(1 - \exp\left\{\frac{1-4^{R}}{\Gamma_{s,d}}\right\}\right) \end{aligned} \tag{3.60}$$

Likewise, with random selection strategy, (3.59) becomes

$$\begin{aligned} P_o &= \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,d}}\right\}\right) \left(1 - \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r}\right) \\ &\quad + \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r} \left(1 - \exp\left\{\frac{1-4^{R}}{\Gamma_{s,d}}\right\}\right) \\ &\quad - \left(1 - \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r}\right) \int_0^{4^{R/\alpha}-1} \frac{1}{\Gamma_{s,d}} \exp\left\{-\frac{\gamma}{\Gamma_{s,d}}\right\} \exp\left\{\frac{1}{\Gamma_{r,d}}\left(1 - \frac{4^{R/\overline{\alpha}}}{(1+\gamma)^{\alpha/\overline{\alpha}}}\right)\right\} d\gamma \end{aligned} \tag{3.61}$$

For asymptotically high and low relay-destination SNRs, (3.60) and (3.61) will reduce to

$$\lim_{\Gamma_{r,d}\to\infty} P_o = \left(1 - \exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r} \left(1 - \exp\left\{\frac{1-4^{R}}{\Gamma_{s,d}}\right\}\right) \tag{3.62}$$

Likewise,

$$\begin{aligned}\lim_{\Gamma_{r,d}\to 0} P_o &= \left(1-\exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,d}}\right\}\right)\left(1-\left(1-\exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r}\right)\\ &\quad+\left(1-\exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r}\left(1-\exp\left\{\frac{1-4^{R}}{\Gamma_{s,d}}\right\}\right)\end{aligned} \tag{3.63}$$

3.5.4 Macrodiversity Multihop

Finally, we consider a special example of the multiple-relay channel, where the source and destination are separated far apart such that there is no direct path available between them, i.e. $\gamma_{s,d} = 0$. This particular network setup, termed macrodiversity multihop, was studied in [71] for a frequency-shift keying system. Here, we will consider its information-theoretic performance under the relay clustering assumption. This provides some insight into the relative importance of the direct source-destination link.

The OEP of this macrodiversity multihop system is found as:

$$P_o = \sum_{k=1}^{K_r} Pr\left\{|D(s)| = k+1\right\} Pr\left[\left\{\overline{\alpha}C\left(\gamma_{r,d}\right) \leq R\right\} \mid \left\{|D(s)| = k+1\right\}\right] + Pr\left\{|D(s)| = 1\right\} \tag{3.64}$$

With optimal selection strategy, (3.64) becomes

$$P_o = \left(1-\exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}+\frac{1-4^{R/\overline{\alpha}}}{\Gamma_{r,d}}\right\}\right)^{K_r} \tag{3.65}$$

Likewise, with random selection strategy, (3.64) becomes

$$P_o = 1-\exp\left\{\frac{1-4^{R/\overline{\alpha}}}{\Gamma_{r,d}}\right\}+\left(1-\exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r}\exp\left\{\frac{1-4^{R/\overline{\alpha}}}{\Gamma_{r,d}}\right\} \tag{3.66}$$

Asymptotically, (3.65) and (3.66) will reduce to

$$\lim_{\Gamma_{rd}\to\infty} P_o = \left(1-\exp\left\{\frac{1-4^{R/\alpha}}{\Gamma_{s,r}}\right\}\right)^{K_r} \tag{3.67}$$

Likewise,

$$\lim_{\Gamma_{rd}\to 0} P_o = 1 \tag{3.68}$$

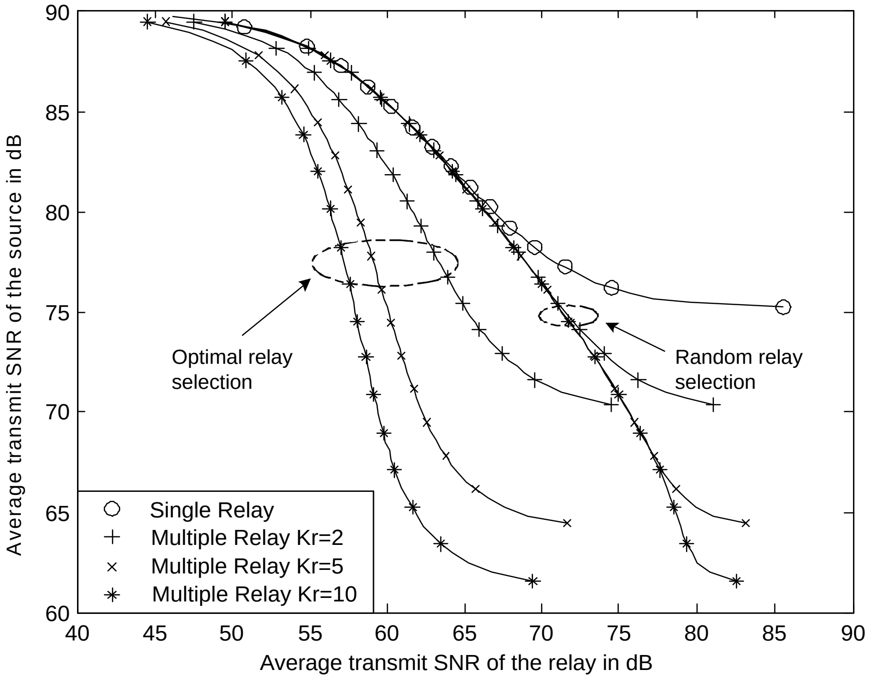

Figure 3.9: Minimum transmit SNR to achieve an end-to-end outage event probability of 10^{-2} with decode-forward relaying in the multiple-relay channel. Optimal selection strategy uses the decoding relay with the best SNR to the destination, while random selection strategy randomly selects a relay from the decoding set.

3.5.5 Performance Comparison

For analytical convenience, Fig. 3.9-3.11 only compare the performance of different multiple-relay network protocols when $\alpha = 0.5$. Extension to variable α is possible. The contours show the minimum combination of source and relay transmit SNR (Γ_s, Γ_r) required to achieve an end-to-end OEP of 10^{-2}. In particular, Fig. 3.9 investigates the performance of different selection strategies for decode-forward relaying with a variable number of relays. In this figure the source only transmits the first block while the second block is transmitted by a relay (provided that at least one received it). Observe that as the number of relays increases, optimal selection strategy is able to achieve significant savings in power at both the source and the relay. However random selection strategy only conserves power at the source. Due to relay clustering, both strategies benefit from a receive diversity effect which helps to improve the energy efficiency of the source. However, energy efficiency at the relay can only be improved by applying optimal selection strategy to obtain transmit diversity (random selection doesn't have any transmit diversity benefit).

Fig. 3.10 studies the advantages of applying source adaptive protocol SA-CC in the multiple-relay channel. Observe that the advantage of SA-CC relative to DF-CC diminishes as the number of relays increases. Fig. 3.11 illustrates the influence of the direct source-destination path on the

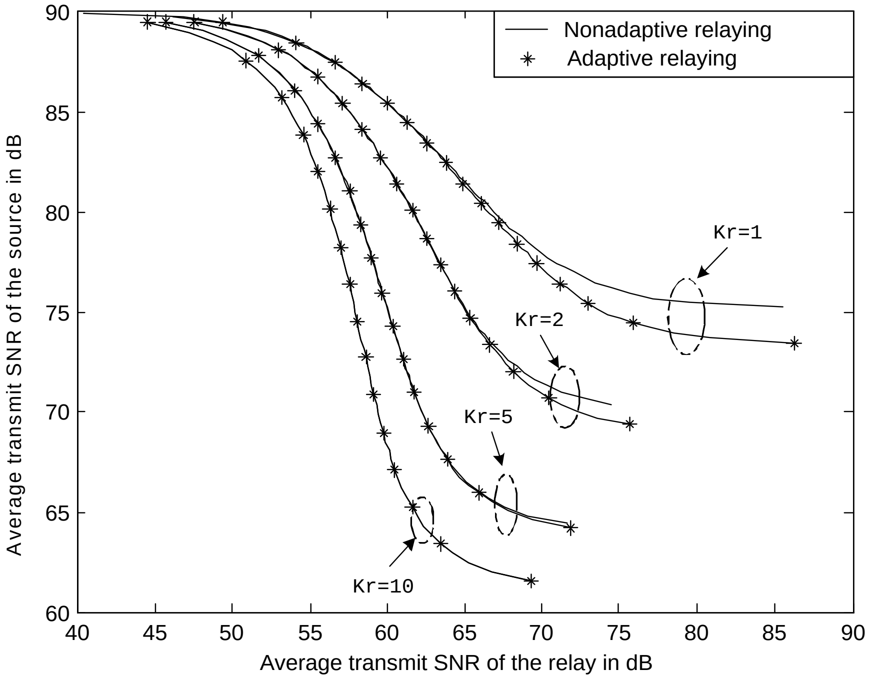

Figure 3.10: Performance comparison of source adaptive protocol SA-CC vs. nonadaptive decode-forward relaying in the multiple-relay channel applying optimal selection strategy. Contours show the minimum transmit SNR to achieve an end-to-end outage event probability of 10^{-2}.

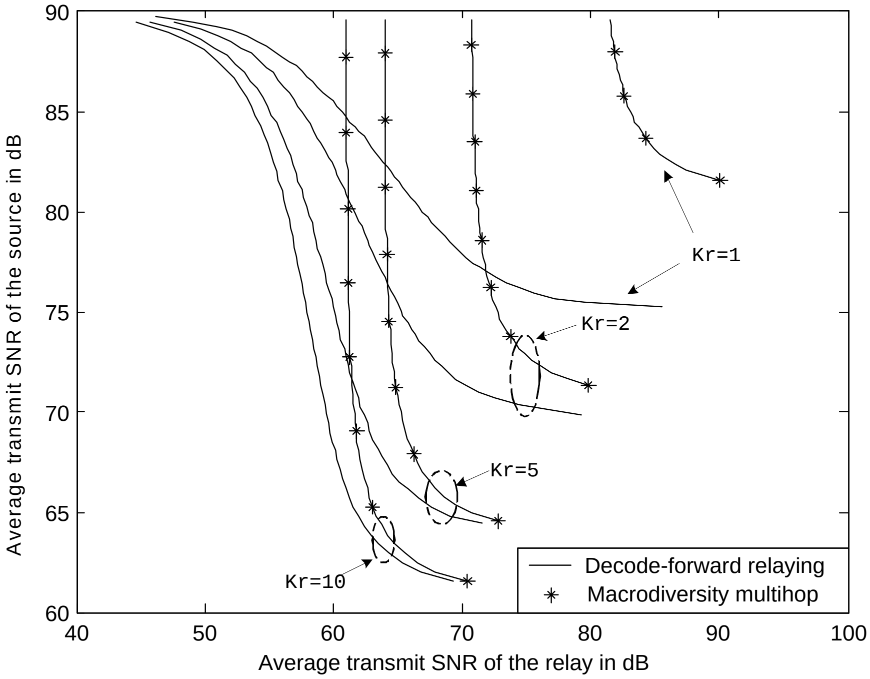

Figure 3.11: Performance comparison of macrodiversity multihop (no source-destination path) vs. nonadaptive decode-forward relaying in the multiple-relay channel applying optimal selection strategy.

performance of the multiple-relay channel. Notice that when the source transmit power is high and there is a small number of relay nodes, the existence of a direct source-destination path significantly improves the energy efficiency. However, as the number of relays increases, the performance gap between macrodiversity multihop (no direct source-destination path) and decode-forward relaying becomes marginal, especially with low source power. Although these conclusions are drawn based on the multiple-relay channel under optimal selection strategy , they are also valid for random selection strategy.

3.6 Summary

In this chapter, information-theoretic limits were found for rate constrained wireless relay channels involving two orthogonally transmitted blocks. We first discussed the capacity of the constrained single-relay channel with different non-adaptive relaying protocols and derived expressions for outage probability. Then, we proposed several adaptive protocols, attempting to combine the potential advantages of the different non-adaptive protocols in block fading scenarios. By applying the analytical results of adaptive relaying protocols, we studied user cooperative coding from the adaptive relaying perspective, and characterized the information theoretic limit of user cooperative coding. In addition, performance limits of the rate constrained channels with multiple relays were found and compared against an alternative relaying scheme which does not require a direct path from source to destination (macrodiversity multihop). It was shown that the relative importance of the direct source-destination link diminishes at low SNR as the number of relays increases.

If feedback from the destination is available in the relay network, then relays only need to transmit when the destination does not correctly decode the original block broadcast by the source. This suggests the potential for further energy savings by combining the proposed strategies with higher layer mechanisms, such as Automatic Repeat reQuest (ARQ). In particular, our current study on relay channels could be further extended to analyzing relay networks that comprise multiple relay nodes and allow for multiple blocks of transmission where ARQ based relaying protocols is used to guide the message transmission. One striking difference is that relay networks involves inter-relay communication. This brings up a fundamental tradeoff between the additional spatial diversity benefit and the MAI penalty due to the existence of multiple relays transmitting simultaneously.

Chapter 4

Hybrid ARQ Protocols for Rate Constrained Relay Networks

4.1 Introduction

In the previous chapter, we investigated clusters of random access relay networks under a particular rate constraints $M = 2$. With $M = 2$, the network clusters are reduced to a simple orthogonal relay channel, where only two transmissions are allowed for each message. As we observed in the previous chapter, even under a small rate constraint, significant energy savings are possible by implementing spatial diversity in a block fading channel. However, the major limitation is that the diversity benefit as well as the energy efficiency of the system is rather restricted by the small rate constraints imposed on the network configuration. Although, in the orthogonal multiple relay channel, the benefit of selection diversity increases along with the number of relay nodes in the channel which helps to reduce the total energy consumption in the network, its achievable diversity gain relies heavily on a highly coordinated relaying protocol among source and multiple relay nodes, whose design and implementation will quickly become unwieldy in a large scale network. To further exploit spatial diversity with reasonable implementation complexity in relay networks under larger rate constraints, we proposed several Automatic Repeat Request (ARQ) based protocols to guide the message transmission through the network. For instance, if feedback from the destination is available in the relay network cluster, then relays only need to transmit when the destination does not correctly decode the original block broadcast by the source. The energy saving by using this mechanism is apparent, although message delay becomes a major concern in the practical implementation. Essentially, the protocol design boils down to the fundamental tradeoff between energy efficiency and delay related throughput efficiency in rate constrained relay networks.

In this chapter, we intend to directly confront the issues of network energy efficiency by the design and analysis of effective relaying protocols which operate across several network layers within a cluster of cooperating devices. The focus of the study is to explore the tradeoff between energy efficiency, throughput, and delay in rate constrained relay networks. The remainder of this chapter is organized as follows: Section 4.2 proposes several interference-free hybrid-ARQ based relaying protocols that effectively transport information and conserve energy within the network. Section 4.3 uses Monte Carlo integration to analyze the asymptotic performance, i.e. throughput as well as energy efficiency, of these relaying protocols under various system constraints and network topologies. Section 4.4 investigates some practical implementation issues in the relay network. Finally, we draw conclusions in section 4.5.

4.2 Hybrid-ARQ Based Protocols for Relay Networks

Without ARQ protocols, the cluster will transmit all M blocks of the codeword before moving on to the next message. This is wasteful of network resources, as often the destination may be able to successfully decode after receiving some earlier block $m < M$. On the other hand, with ARQ the cluster will only transmit new blocks of the codeword until one of the following occurs: (1) the destination successfully decodes the message and signals back with a positive acknowledgement (ACK), which we assume for the sake of exposition is conveyed over an error- and delay-free feedback channel; (2) all M blocks have been transmitted, $m = M$; or (3) a maximum latency has been exceeded, $s > D$ (M and D constitute a *rate constraint* and a *delay constraint*, respectively). The relays need not transmit ACKs of their own, although relay-ACKs may prove to be convenient for some implementations.

First consider how hybrid-ARQ can be used to effectively determine the set $\mathcal{K}(s)$ of transmitters. Let $\mathcal{D}(s)$ denote the set of nodes with knowledge of the codeword at the start of slot s; we call $\mathcal{D}(s)$ the *decoding set* and its members *decoding nodes* (The decoding set concept was proposed in [46] for a nonadaptive system and thus with no dependence on the slot s.). Under decode-and-forward relaying, only decoding nodes may transmit, and thus $\mathcal{K}(s) \subseteq \mathcal{D}(s)$. Initially, the decoding set contains only the source, $\mathcal{D}(s_1) = \{Z_s\}$. After the first block and at the start of the m^{th} block, the decoding set will contain the source plus all relays that have previously accumulated enough information to decode successfully, i.e. $\mathcal{D}(s_m) = \{Z_s, Z_k : I_k[m-1] > R\}$. Once a relay is added to the decoding set, it is never taken out, so $|\mathcal{D}(s)| \geq |\mathcal{D}(s-1)|$, where $|\mathcal{X}|$ is the cardinality of set $\mathcal{X}$. Once a node is in the decoding set, it no longer needs to listen and therefore does not expend any more energy receiving and processing additional blocks of the codeword.

The source begins by broadcasting the first block during the first slot ($s_1 = 1$). The destination can decode the message if $I_d[1] > R$ and, if successful, will broadcast an ACK. Otherwise, a retransmission will be necessary. After the source's initial broadcast, some of the relays may have successfully decoded the transmission, namely those for which $I_k[1] > R$. These decoding relays are included in $\mathcal{D}(2)$. During the next transmission slot $s \geq 2$, *any* node in $\mathcal{D}(2)$ can transmit the second block of the codeword. But which?

We assume that each node in $\mathcal{D}(s)$ makes a *localized* decision to transmit, although the decision could be influenced by any knowledge the node may possess regarding the composition of $\mathcal{D}(s)$ (i.e. what other relays have decoded successfully?), its transmit SNR (instantaneous or average), and the transmit SNR of other nodes in $\mathcal{D}(s)$. Let $p_k[s]$ be the probability that Z_k transmits during slot s given that Z_k is in the decoding set, $p_k[s] = Pr\{Z_k \in \mathcal{K}(s) | Z_k \in \mathcal{D}(s)\}$. In all our protocols, we assume random cluster activity, in the sense that S_m is a random set of time slots. However, during active slots (i.e. $s = s_m \in S_m$), the protocol could be either random (arbitrary $p_k[s_m]$) or deterministic ($p_k[s_m] \in \{0, 1\}$ and chosen *a priori*).

4.2.1 Interference-Free Relaying Protocols

We initially consider three genie-aided methods for selecting which, if any, node transmits during a particular slot. At most one node will transmit at a time, hence $|\mathcal{K}(s_m)| \leq 1, \forall m$. Since there is no intra-cluster interference, we term these strategies "interference-free relaying". During a particular slot s, the probability of a transmission within the cluster is p_t, which we assume is constant for all slots. Once it is determined that a node within the cluster will transmit, the choice of *which* node actually transmits is a deterministic function of the SNRs (Since there is no intra-cluster interference in this protocol, the SINR and SNR are identical (assuming inter-cluster interference is treated as additional noise).

Max-Relaying Protocol

Recall that in chapter 3, we have briefly studied max-relaying scheme in orthogonal relay channels under rate constraint $M = 2$, where the relay node with the best instantaneous channel will forward the message. Here max-relaying scheme will be generalized into an interference-free relaying protocol by further incorporating hybrid-ARQ technique, such that it could be applied to relay networks with larger rate constraint. Similar to the max-relaying in chapter 3, we choose the node to transmit the m^{th} block to be the one whose channel to the destination has the highest *instantaneous* SNR, i.e. $\mathcal{K}(s_m) = \{Z_k : k = \arg\max_i \gamma_{i,d}[m]\}$. While the exact term for this protocol is 'interference-free relaying using maximum instantaneous SNR', we will continue call it max-relaying for short. Note that this protocol requires knowledge of the *current* SNR of every relay in the decoding set to the destination; it does not, however, require knowledge of *future* SNRs.

Average-Relaying Protocol

Alternatively, the node with the highest *average* SNR to the destination could be selected, i.e. $\mathcal{K}(s_m) = \{Z_k : k = \arg\max_i \Gamma_{i,d}[m]\}$. We term this protocol 'interference-free relaying using maximum average SNR' or average-relaying for short. This relaying scheme is closely related to the 'Geographic Random Forwarding' (GeRaF) routing protocol proposed in [7] where the node geographically closest to the destination tends to forward the message. However, the key distinction is that GeRaF operates under the assumption that all nodes in range will always be able to decode, while average-relaying takes into account the fading and interference which could make the transmission to an in-range node fail. Furthermore, it also takes into account the hybrid-ARQ protocol, which allows us to study the tradeoff between transmit power and network delay. However, with average relaying, the network empty zone problem is hard to overcome. An empty zone is a region of the network where there is no active relay node in it. When the current relaying node has encountered an empty zone and there is no other node closer to the destination in its broadcast region, then according to the average-relaying protocol, the message will get stuck, i.e. no more forward progress can be made. There are two possible ways to mitigate this problem. On the one hand, we could introduce mobility to the network so that some nodes will eventually move into the region and fill up the empty zone. Alternatively, we could modify the protocol in such a way that whenever a message encounters an empty zone, it is able to take a detour. We propose solving this problem by introducing a probabilistic transmission mechanism into the average-relaying protocol where each potential relay node is assigned a transmission probability based on its geographic location. In that case, the node closest to the destination will transmit with higher probability but not always. Therefore once the message encounters an empty zone, some other nodes (not so close to the destination) will also get a chance to transmit, so that eventually the message is able to reach the destination.

Random-Relaying Protocol

Finally, we could randomly pick a node in the decoding set to transmit the m^{th} block regardless of its channel information. This results in 'interference-free relaying with random selection' or random-relaying for short. Notice that random-relaying requires neither channel instantaneous SNR nor geographic information, rather it only relies on decoding set information to relay the message.

The max-relaying protocol ensures that the destination accumulates a maximum amount of mutual information during each time slot. This is optimal if the goal is to maximize the probability that the destination is able to decode at the end of that particular slot. However, in the presence of multiple relays, this mechanism does not necessarily ensure maximum system throughput or energy efficiency. This is because at very low SNR and with multiple relays, it is often more important to maximize the probability that the *next* relay along a chain can decode, rather than the probability that the destination (which is at the end of the chain) can decode. Thus, as we will show later, average-relaying can often outperform max-relaying because average SNRs are more relevant than instantaneous SNRs whenever there is more than just one transmission (hop) remaining before the destination decodes (as in the case of low SNR and many relays).

4.2.2 Relaying vs. Multihop

Consider how relay networks differ from conventional multihop networks. With multihop, the message must flow through the cluster following a series of direct peer-to-peer connections that are determined *a priori* by a routing algorithm. Without loss of generality, we assume that under multihop the message must flow through *all* K_r relays before reaching the destination and that the relays are indexed in the order that they are used. Under multihop, only the *next* node $Z_{|\mathcal{D}(s)|+1}$ not yet in the decoding set receives the transmission, while with relaying *all* nodes not yet in the decoding set $\{Z_k \notin \mathcal{D}(s)\}$ receive. With multihop, all relays in the cluster must eventually decode the message, $\mathcal{D}(s_M + 1) = \mathcal{N}$, but with relaying it is irrelevant which relays have successfully decoded; all that matters is if the destination was able to decode successfully, i.e. $Z_d \in \mathcal{D}(s_M + 1) \subseteq \mathcal{N}$. With the proposed relaying protocols, relays that are repeatedly in an outage are bypassed, thereby eliminating potential bottlenecks. Furthermore, a network-layer protocol is not needed to preselect the transmission path, rather the "path" selection is embedded into the ARQ mechanism (although we argue that the term *path* becomes meaningless). Also, power/range control becomes less important in a relaying network. In a multihop network, if the transmit power is too high, then the extra energy is wasted. However, if the power is set too high in a relay network then intermediate relays will simply be "leapfrogged" and therefore won't need to be used.

4.3 Relaying under Infinite Delay/Rate Constraint

Hybrid-ARQ systems are typically characterized in terms of throughput and delay statistics. However for many ad hoc networking applications, a more critical metric is the energy efficiency $\mathcal{E}_b$, namely, the average *cumulative* energy expended by the network to convey a correct bit [72], since most low-cost network devices have limited energy reserve. Note that in relaying networks, this energy consumption is often spread among several nodes, i.e. all those that participated in relaying the message. In the following example, we consider the relationship between such performance metrics as throughput, delay, and energy efficiency of equidistant line networks. In particular, we intend to develop a unified analysis based on the renewal-reward theorem [56] to study the fundamental tradeoff among those metrics under interference-free relaying protocols.

Consider an equidistant line network with a 100 m source-destination separation and K_r relays equally spaced along the line connecting the source and destination. We assume the node activity factor is $p_t = 0.1$, the block/burst rate $R = 1$, transmit frequency $f_c = 2.4$ GHz, path loss coefficient $\mu = 3$, and reference distance $d_o = 1$ m. The block fading coefficients are Rayleigh distributed and independent. All nodes transmit with equal energy per symbol $\mathcal{E}_k[m] = \mathcal{E}_s \quad \forall k, m$. Monte Carlo integration/simulation was used to evaluate both throughput and energy efficiency.

In order to apply the renewal-reward theory to those relaying protocols as well as conventional multihop, we first define the following random variables:

$\mathcal{R}$: A random reward, equal to R if the packet is successfully decoded by the destination and zero otherwise.

$\mathcal{T}$: The time (in number of slots) spent attempting to transmit an arbitrary message (until either success or until the delay/rate constraints expire).

$\mathcal{M}$: The total number of blocks transmitted for an arbitrary message until success or the constraints expire.

For analytical convenience, we assume infinite delay/rate constraint, e.g. $D \rightarrow \infty$ and $M \rightarrow \infty$. System performance under finite delay/rate constraint will be covered later in this chapter. It is straightforward that the infinite delay/rate constraint assumption reduce the complexity involved in the analysis. For instance, the random reward $\mathcal{R}$ is reduced to a constant $\mathcal{R} \equiv R$, since the destination will always eventually decode the message.

4.3.1 Throughput

In [21], the renewal-reward theorem of [56] was used to determine bounds on the throughput of a multi-source network of concurrent point-to-point links (no relaying) in the presence of Rayleigh block fading and Gaussian interference.

Applying the same theorem, the system throughput is

$$\eta = \frac{E[\mathcal{R}]}{E[\mathcal{T}]}, \tag{4.1}$$

in units of messages per slot.

We further notice that interference-free relaying protocols ensure that at most one block is transmitted during each slot. Therefore, when each node has a duty cycle p_t, the average delay of the interference-free protocols becomes

$$E[\mathcal{T}] \quad = \quad 1 + \frac{E[\mathcal{M}] - 1}{p_t} \tag{4.2}$$

where $E[\mathcal{M}]$ is the average number of blocks transmitted until the destination can correctly decode the message. Note that when there is always a node transmitting during every slot ($p_t = 1$), $E[\mathcal{T}]$ is reduced to $E[\mathcal{M}]$.

Although messages are sequentially passed through peer-to-peer links in multihop networks, without delay or rate constraints, these links are identical in the sense of throughput and average delay (except for the first hop where the first transmission from the source is deterministic). In particular, the average delay of multihop for the i^{th} hop is

$$E[\mathcal{T}_i] = \begin{cases} E[\mathcal{M}_{SH}]/p_t, & \text{for } i > 1 \\ 1 + (E[\mathcal{M}_{SH}] - 1)/p_t, & \text{for } i = 1 \end{cases} \tag{4.3}$$

where $E[\mathcal{M}_{SH}]$ denotes the expected number of blocks transmitted for an arbitrary message in a single-hop (direct-connection) network. Accordingly, the average delay of a multihop system over an equidistant line network is the accumulation of delay components at each individual hop,

$$E[\mathcal{T}] = 1 + ((K-1)E[\mathcal{M}_{SH}] - 1)/p_t \tag{4.4}$$

Since $\mathcal{R} \equiv R$, when $D \to \infty$ and $M \to \infty$, by plugging (4.2) and (4.4) into (4.1), the throughput of interference-free relaying and multihop over the line network is

$$\eta \quad = \quad \begin{cases} R/(1 + (E[\mathcal{M}] - 1)/p_t), & \text{for relaying} \\ R/(1 + ((K-1)E[\mathcal{M}_{SH}] - 1)/p_t), & \text{for multihop} \end{cases} \tag{4.5}$$

Fig. 4.1 shows the throughput of $K_r = 1$ and $K_r = 10$ equidistant line networks without rate/delay constraints as a function of burst transmit SNR. At low SNR, multihop actually outperforms max-relaying. This result seems counter-intuitive, since with relaying, information accumulates at all nodes after each burst, while with multihop the information only accumulates at the next node along the line. However, at low SNR only the next node along the line accumulates a meaningful quantity of information; the amount of information accumulated at the downstream nodes is negligible due to path loss effect. Thus relaying is closely approximated by multihop at low SNR. As mentioned earlier, max-relaying focuses on maximizing the information received by the destination after each burst rather than the information of the next node along the line, which as indicated by the results is a suboptimal strategy at low SNR. However, average-relaying does indeed outperform both multihop and max-relaying, since it tends to maximize the average information received by the next node along the line. Meanwhile, interference-free random-relaying

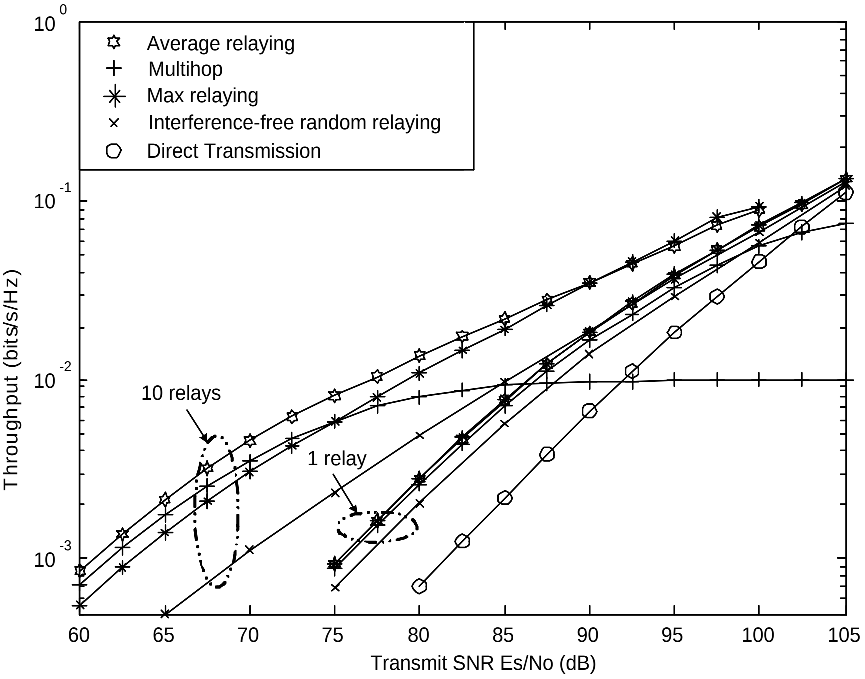

Figure 4.1: The throughput in the absence of rate and delay constraints as a function of burst transmit SNR for a $K_r = \{1, 10\}$ relay line network. The source and distance are separated by 100 meters.

has a much lower throughput, due to its random selection of node transmission which does not maximize the information received by either the destination or the next node along the line. At moderate to high SNR, all three relaying protocols significantly outperform multihop. More interestingly, max-relaying now outperforms average-relaying. This is because at high SNR, there is a high probability that if the initial source transmission failed, the second transmission will succeed. Thus the max-relaying strategy of maximizing information received by the destination during each block helps to improve the overall throughput.

In general, we observe that although the throughput of all protocols are non-decreasing functions of transmit SNR, the throughput of multihop saturates after a threshold SNR. This is because multihop requires hops through every node, while relaying permits nodes to be skipped. Thus at high SNR, the peer-to-peer transmission of multihop becomes a bottleneck. One could argue that at high SNR, the performance of multihop could be improved by selecting a new route that uses fewer relays. However, the beauty of relaying is that it will do this automatically without needing to adjust the route.

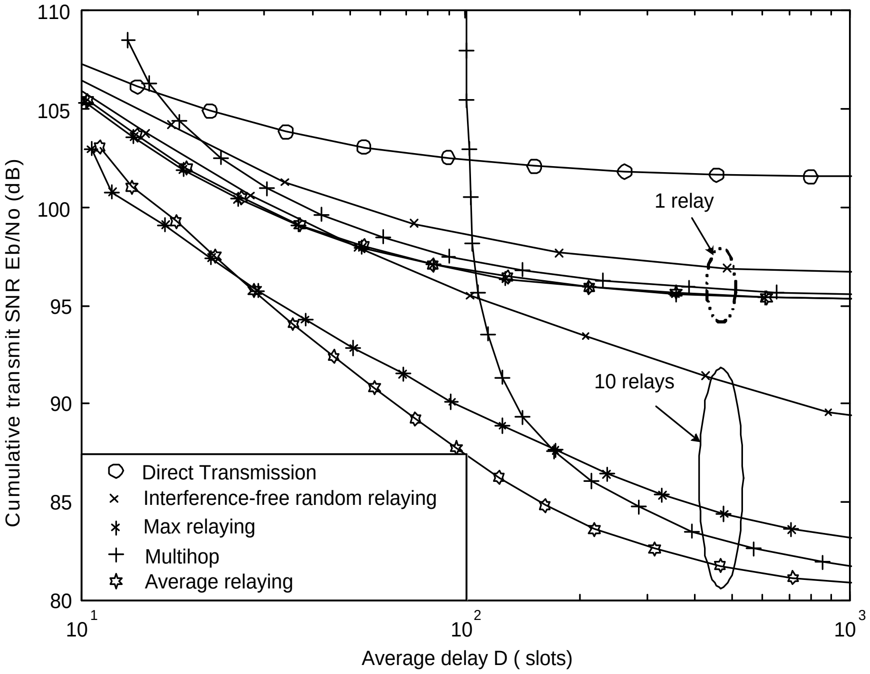

Figure 4.2: The minimum cumulative transmit-only SNR as a function of average delay for a $K_r = \{1, 10\}$ relay line network in the absence of rate and delay constraints. The source and distance are separated by 100 meters.

4.3.2 Energy Efficiency

Transmit Energy Efficiency

As mentioned earlier, energy efficiency is one of the most critical metrics used to evaluate the performance of ad hoc networks. Applying renewal-reward theory, the average *cumulative* transmit energy is

$$\begin{aligned} \mathcal{E}_b &= \frac{\mathcal{E}_s E[\mathcal{M}]}{E[\mathcal{R}]} \\ &= \begin{cases} \mathcal{E}_s E[\mathcal{M}]/R, & \text{for relaying} \\ \mathcal{E}_s (K-1) E[\mathcal{M}_{SH}]/R, & \text{for multihop} \end{cases} \end{aligned} \tag{4.5}$$

where $\mathcal{E}_s$ denotes the transmit energy of every symbol (here we do not allow power control, so all nodes transmit with the same energy). Rather than representing the energy transmitted by any *single* node, $\mathcal{E}_b$ characterizes the energy consumed by the *entire* cluster by enumerating the total number of transmitted blocks per correct message without regard to which nodes transmitted the blocks. For networks without any rate or delay constraints, energy efficiency $\mathcal{E}_b$ is a unique function of burst transmit $\mathcal{E}_s$, block rate R, network topology, and protocol.

Fig. 4.2 shows the cumulative transmit-only SNR ($\mathcal{E}_b/N_o$) as a function of average delay without rate/delay constraints for the four protocols and $K_r = 1$ and 10 equidistant line networks. For each value of average delay and protocol type, there is a unique corresponding $\mathcal{E}_s$. The performance of direct-transmission ($K_r = 0$) is also shown for comparison. As expected, the two interference-free relaying protocols are always more efficient than random-relaying. In particular, for a 10 relay network at an average delay of 1000, average-relaying is about 8 dB more efficient than random-relaying, while max-relaying is about 6 dB more efficient. Although relaying is always more efficient than direct-transmission, multihop performs worse in the region of small average delay (corresponding to high $\mathcal{E}_s$). In order to reduce the average message delay, each node in the network must operate (transmit) at high SNR where multihop reaches its performance bottleneck. As the average delay becomes larger, the energy efficiency of both multihop and relaying protocols will be significantly improved. In general, the transmit energy efficiency is improved at the price of system throughput. This agrees with Caire's assertion that 'the longer we wait the more we gain' [21].

Dissipated Energy Efficiency

While Fig. 4.2 indicates the advantages of *transmit* energy efficiency by using more sophisticated relaying protocols, the benefits of these protocols must be weighed against their costs. Perhaps the most critical issue is that now *all* nodes that are not yet in the decoding set must receive every transmission, as opposed to multihop which requires that only *one* node receives. Thus a fair comparison between relaying and multihop should also account for the energy a node consumes when it *receives* a symbol. For the relaying protocols, the number of nodes that receive the m^{th} block is $K - |\mathcal{D}(s_m)|$ and must be taken into account. Therefore, we can generalize the definition of cumulative energy dissipation to

$$\mathcal{E}_b = \begin{cases} (\mathcal{E}_s E[\mathcal{M}] + \mathcal{E}_r E[\sum_{s \in S_m} (K - |\mathcal{D}(s)|)])/R, & \text{relaying} \\ (\mathcal{E}_s + \mathcal{E}_r)(K-1)E[\mathcal{M}_{SH}]/R, & \text{multihop} \end{cases} \tag{4.6}$$

where $\mathcal{E}_s$ denotes the transmitted energy per symbol and $\mathcal{E}_r$ denotes the energy consumed by the *receiver* to detect and process a symbol.

Notice that with interference-free protocols, the energy consumption does not depend on the node duty cycle p_t, while the throughput is only scaled by a factor of approximately p_t. Thus, an analysis can be carried out by normalizing $p_t = 1$, thereby eliminating the dependence on p_t and allowing for a more direct comparison of other effects, such as the protocol, the number of relays, and the burst transmit power.

One should realize that $\mathcal{E}_r$ is highly implementation dependent. In [73], it is assumed that the energy consumed by receiving and processing a symbol is equal to the energy to transmit it. For multihop, this assumption implies that the energy the cluster consumes conveying a bit is simply

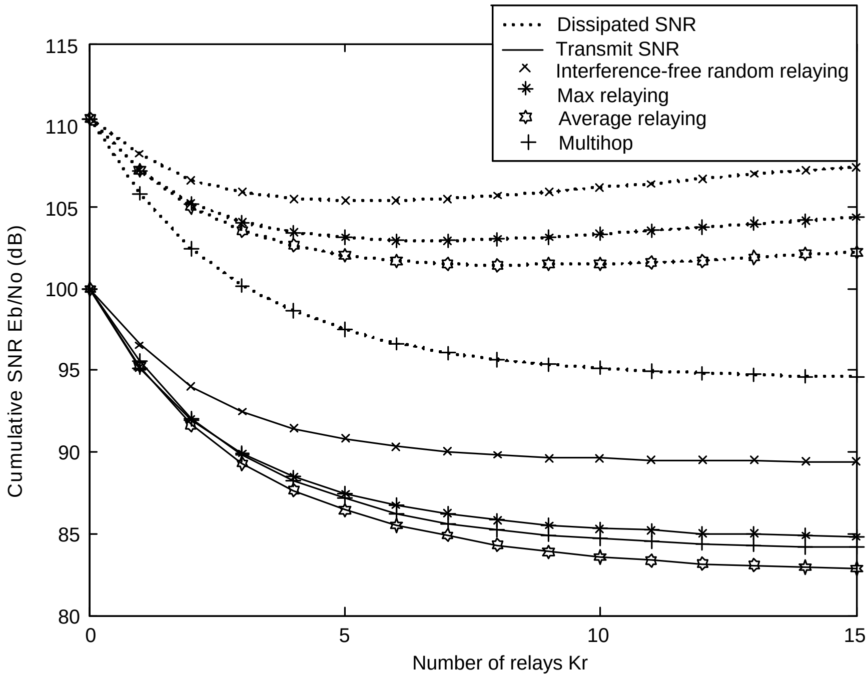

Figure 4.3: The minimum required transmit-only SNR and cumulative SNR for a line network in the absence of rate and delay constraints as a function of the number of relays when the burst transmit SNR is $\mathcal{E}_s/N_o = 70$ dB and $\mathcal{E}_r/N_o = 80$ dB. The source and distance are separated by 100 meters.

doubled. A problem with this assumption is that it gives an unfair advantage to protocols that are more transmit-energy efficient. However, since $\mathcal{E}_r$ is just function of the physical implementation of the receiver, reducing the required transmit $\mathcal{E}_s$ does not mean that the dissipated $\mathcal{E}_r$ will be any lower. A better comparison is to select a typical value for $\mathcal{E}_r$ and use it for all of the protocols. In the following, we set $\mathcal{E}_r/N_o$ equal to a typical value for $\mathcal{E}_s/N_o$, namely $\mathcal{E}_r/N_o = 80$ dB.

Figs. 4.3 and 4.4 show the cumulative dissipated SNR for two burst transmit SNRs as a function of the number of relays under the assumption that $\mathcal{E}_r/N_o = 80$ dB. The results in Fig. 4.3 indicate that at lower SNR, i.e. $\mathcal{E}_s/N_o = 70$ dB, average-relaying is most efficient when $\mathcal{E}_r$ is negligible compared to $\mathcal{E}_s$. However, when $\mathcal{E}_r$ is comparable to or even greater than $\mathcal{E}_s$, multihop is most efficient in terms of the total dissipated energy. The real benefits of relaying are achieved when each block is burst with high SNR. As shown in Fig. 4.4, the transmit-only energy efficiency of each relaying protocol is a monotonically decreasing function of the number of nodes. However, the total dissipated energy efficiency of the relaying protocols begins to rise with an increasing number of relays. The required dissipated energy for multihop also increases with K_r, and this increase is at a much faster rate than that of relaying. Thus, while relaying is desirable under high transmit SNR, the advantage over multihop diminishes when the energy to receive and process a symbol is non-negligible. Furthermore, relaying is less sensitive than multihop to the choice of the number of

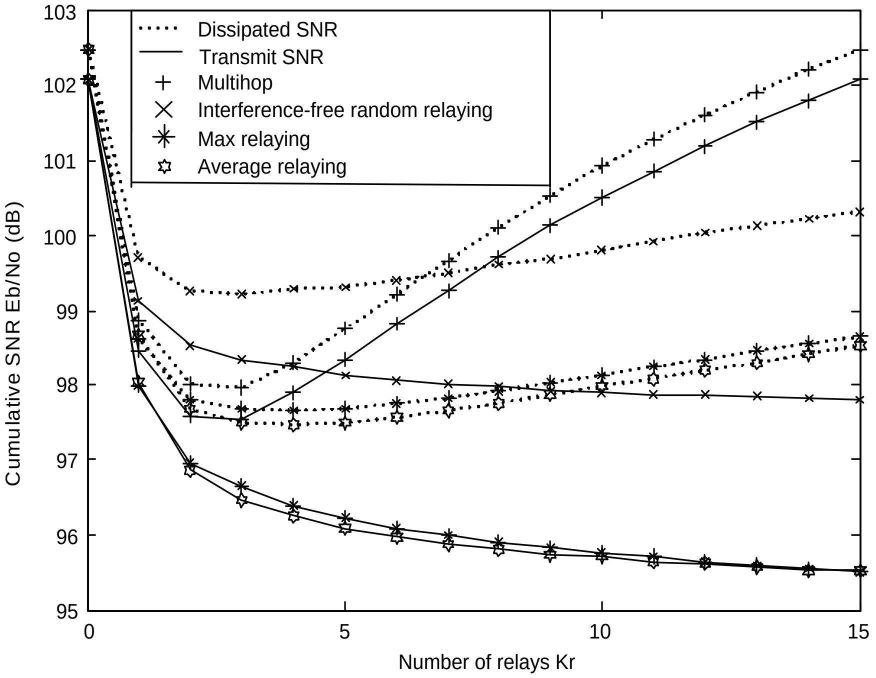

Figure 4.4: The minimum required transmit-only SNR and cumulative SNR for a line network in the absence of rate and delay constraints as a function of the number of relays when the burst transmit SNR is $\mathcal{E}_s/N_o = 90$ dB and $\mathcal{E}_r/N_o = 80$ dB. The source and distance are separated by 100 meters.

relays, thereby relieving the requirements of the routing protocol.

4.3.3 Effects of Network Topology

We have studied an equidistant line network with 10 relay nodes. This is an ideal network layout which has the best performance in the sense of both throughput as well as energy efficiency. However, in practical systems, constructing such network topology may not always be possible. Rather node clustering or random topology are more realistic models for practical ad hoc systems. Therefore, in this subsection, we will investigate the effects of network layout on relaying performance. We notice that average-relaying has the best tradeoff between energy efficiency and delay constraint among the relaying protocols on an equidistant line network, however, max-relaying still may outperform average relaying at certain transmit SNR region. Therefore it is natural to question whether average relaying is always the best protocol in any arbitrary topology? To answer this question, we compared the transmit energy efficiency of max-relaying and average-relaying under five different network clustering topologies. More specifically, consider a relay network with u mini-clusters. Each mini-cluster contains v relay nodes such that the distance among these nodes are much smaller than the source-destination separation. In addition, nodes within each mini-cluster are separated far enough such that the signal pathes of each node are independent. The

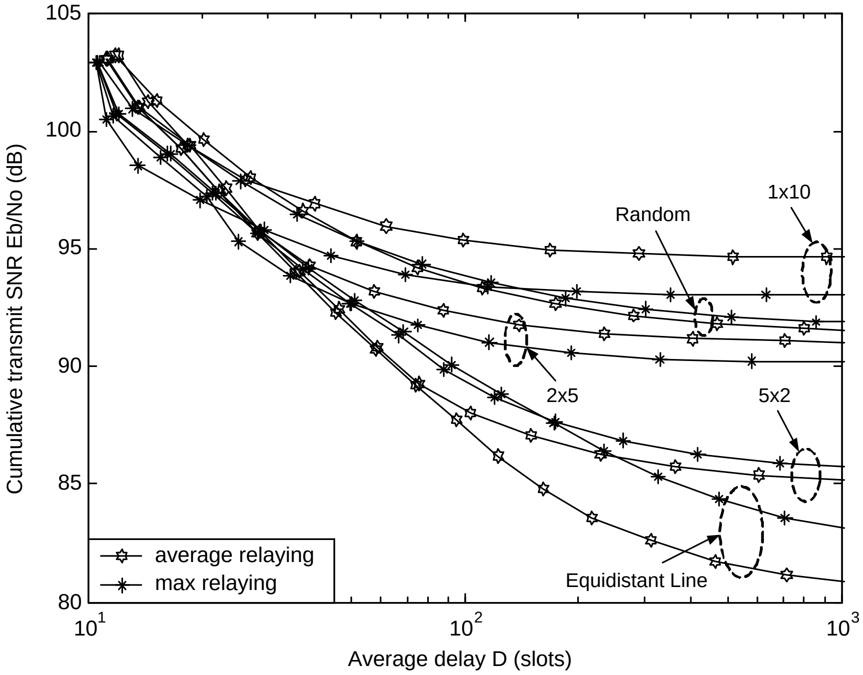

Figure 4.5: The cumulative transmit-only SNR vs. average message delay under different network layouts. The source and distance are separated by 100 meters.

center of each mini-cluster is evenly distributed along a line connecting the source and destination. Therefore, for a relay network with $K_r = 10$, there are 4 possible network configurations: $u \times v = 1 \times 10$, 2×5, 5×2, and 10×1, where 10×1 corresponds to an equidistant line network. As a framework of reference, we also study a random network where 10 relays are randomly placed (with uniform distribution) in a circle of radius 50m. The center of the circle is located halfway between the source and destination. Notice that in Fig. 4.5 the equidistant line (10×1) has the best efficiency among the five network topologies, while 1×10 network is the worst. When the network contains a small number of mini-clusters, max-relaying outperforms average-relaying. As the number of mini-clusters increases, the advantage of max-relaying over average-relaying diminishes until eventually average-relaying outperforms max-relaying. Intuitively, when a network only contains a small number of highly populated clusters, transmit selection diversity will dominate the performance, and thus max-relaying will perform much better than average-relaying (which has no transmit diversity benefit). However, as the number of mini-cluster increases and the number of relays within each cluster decreases, the (micro)diversity benefit diminishes while the path loss effects starts to dominate (macrodiversity). Surprisingly enough, we observe that random network is able to outperform node clustering, i.e. 1×10 network, indicating that in relaying networks macrodiversity is somewhat more important than microdiversity benefit. This conclusion is also compatible with the observation that the performance of networks improve as the number of cluster

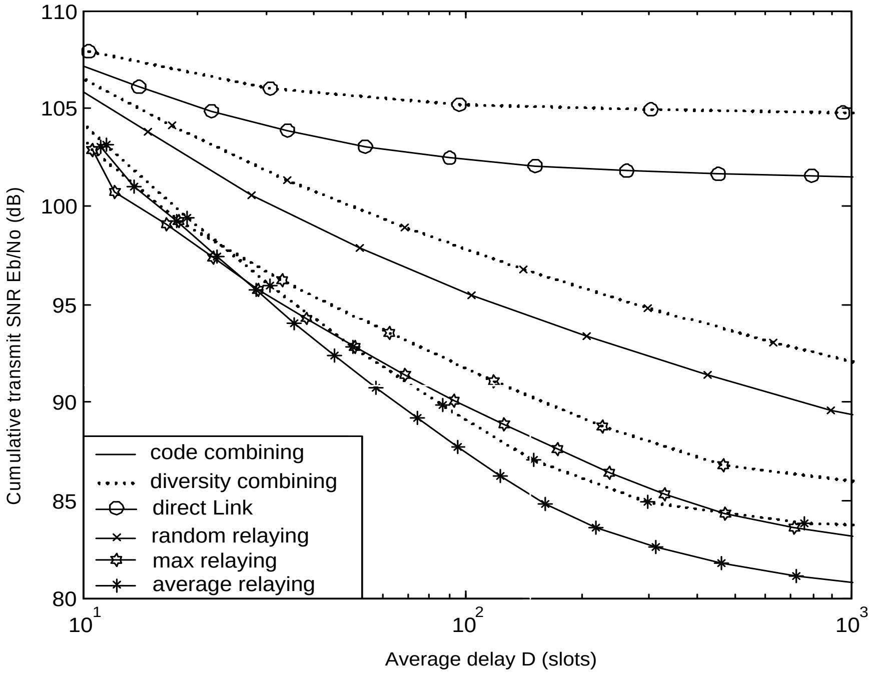

Figure 4.6: The transmit-only energy efficiency of diversity combining schemes is consistently 3 dB worse than that of code combining in a equidistant line network with $K_r = 10$. The source and distance are separated by 100 meters.

in the network increases.

4.3.4 Diversity Combining vs. Code Combining

In the last chapter, we investigated the achievable performance of a special class of orthogonal relay networks, e.g. $M = 2$ rate constrained orthogonal relay channel, under different relaying schemes. We notice that relaying with incremental redundancy and code combining is superior to repetition coding and diversity combining. Under small rate constraints, code combining only causes modest complexity increase over diversity combining in the orthogonal relay channels. However, as the rate constraints increases, the complexity of code combining increases much faster than diversity combining. Recall that with code combining, each device needs to accumulate enough entropy before it could relay the message, e.g. $I_j[m] = \sum_m \frac{1}{2}\log_2(1 + \gamma_j[m])$ [21]. Alternatively, under diversity-combining each device accumulates SINR, i.e. $I_j[m] = \frac{1}{2}\log_2(1 + \sum_m \gamma_j[m])$ [21]. Notice that $\sum_m \frac{1}{2}\log_2(1+\gamma_j[m]) \geq \frac{1}{2}\log_2(1+\sum_m \gamma_j[m])$ due to the convexity of $\log_2(\cdot)$, therefore diversity combining is always inferior to code combining. Essentially, it is a tradeoff between performance and complexity. In Fig. 4.6 and 4.7, we compare the performance of relaying networks with different hybrid-ARQ schemes under relatively large rate constraint, i.e. $M >> 2$. We observe that in both a random network and an equidistant line network, diversity combining is consistently

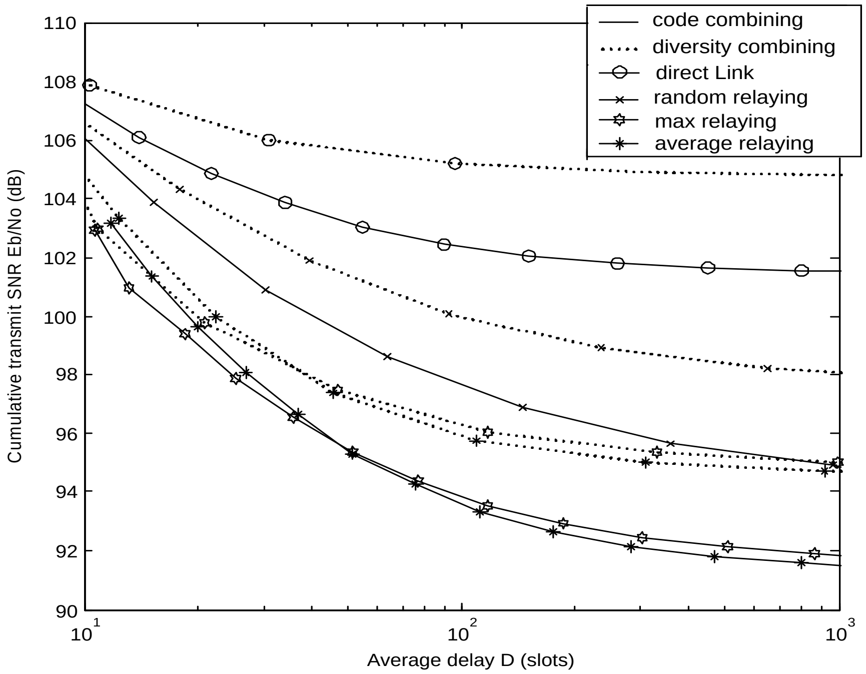

Figure 4.7: In a random network of $K_r = 10$, the transmit-only energy efficiency of diversity combining schemes is also consistently 3 dB worse than that of code combining. In the random network, the source and distance are separated by 100 meters, and 10 relays are randomly placed in a circle of radius 50m. The center of the circle is located halfway between the source and destination.

3 dB inferior to code combining under large rate constraint (large delay). However, as the rate constraint decrease, the advantage of transmit-only energy efficiency in code combining becomes marginal. If we further take into account the receiver energy consumption, diversity combining becomes more attractive in the sense that code combining consumes more receiver energy than diversity combining, possibly due to ML decoder structure or its close approximate, i.e. iterative decoder. Therefore, diversity combining is still desirable in combating complexity in hybrid-ARQ based relaying protocols.

4.4 Implementation Issues

We investigated ideal relaying protocols and their asymptotic performance. However they are relatively hard to implement in practical systems. In particular, interference-free relaying protocols request a highly coordinated global arbitration scheme to avoid collision among multiple devices which might increase the traffic overhead and/or significantly complicate the protocol design. Another practical application issue is finite rate/delay constraints, where each message has certain transmission deadlines, i.e. rate constraint M and delay constraint D. Whenever deadline expires, the corresponding message will be discarded and counted as an outage. Different finite delay/rate constraints will significantly affect the performance of relaying protocols. In this section, we will study these practical issues and propose corresponding solutions.

4.4.1 Finite Delay/Rate Constraint

A striking difference between finite delay/rate constraint and its infinite counterpart is the introduction of outage probability P_o in the throughput and energy efficiency characterization, since when certain transmission deadlines expire, a system outage is counted. In particular, we notice that when D and M are finite, $E[\mathcal{R}] = R(1-P_o)$ where P_o is the outage probability at the destination after the last block has been transmitted, i.e. $P_o = P_d[m]$ where $m = \min(M, \arg\max_m \{s_m \leq D\})$. Applying renewal-reward theory, the average system throughput becomes

$$\eta = \frac{R(1-P_o)}{E[\mathcal{T}]}. \tag{4.7}$$

Likewise, the *cumulative* transmit energy is

$$\mathcal{E}_b^{tx} = \frac{\mathcal{E}_s E[\mathcal{M}]}{R(1-P_o)} \tag{4.8}$$

and the *cumulative* dissipated energy is

$$\mathcal{E}_b = \frac{\mathcal{E}_s E[\mathcal{M}] + \mathcal{E}_r E[\sum_{s\in S_m}(K - |\mathcal{D}(s)|)]}{R(1-P_o)} \tag{4.9}$$

where $S_m = \{s_1, s_2, ..., s_m\}$ denotes a sequence of slots dedicated to relay a particular message, $m = \min(M, \arg\max_m \{s_m \leq D\})$. Notice that $\mathcal{E}_b^{tx}$ denotes the transmit only energy for each bit, while $\mathcal{E}_b$ denotes the total energy dissipation per bit including the transmitter energy and the receiver energy.

In the simulation study to characterize performance under finite delay/rate constraint, we assume network traffic $p_t = 0.1$ so that these results could be directly compared to the infinite delay/rate cases.

Fig. 4.8 and 4.9 show the system throughput and energy efficiency of a line network with 10 relays under two different burst transmit SNRs respectively. Notice that other than three interference-free relaying protocols, we also show the performance of probabilistic random relaying in both figures. The probabilistic random relaying protocol will be discussed in details in section 4.4.2. In general, it simplifies the protocol implementation by incorporates probabilistic transmission into interference-free random relaying protocol. At low SNR, e.g. 70 dB, multihop performs very close to the max-relaying and average-relaying protocols in the sense of both throughput and energy efficiency. Multihop even outperforms max-relaying. On the other hand, at high SNR, multihop has the worst performance. Similar observations were made in Fig. 4.1 and 4.2 where relaying is carried out under infinite delay/rate constraints. Comparing Fig. 4.8 and 4.9 with Fig. 4.1 and 4.2, we further notice that system throughput as well as energy efficiency could be significantly affected by delay constraints D. In particular, at 70 dB, when $D \leq 200$, the achievable throughput as well as energy efficiency of any particular transmission protocol (including relaying and multihop protocols) is much worse than its average throughput under infinite delay/rate constraints. While at relative high transmit SNR, e.g. 100 dB, both throughput and energy efficiency are rather insensitive to the delay constraints. More importantly, we realize that under certain delay/rate constraint, energy efficiency is not a monotonically increasing or decreasing function of transmit SNR. For instance, in Fig. 4.9, when $D = 100$, at 100 dB transmit SNR energy efficiency of all protocols is much better than their performance at 70 dB SNR. On the other hand, if $D = 1000$, their energy efficiency at higher transmit SNR is actually much worse than when they are transmitting with lower SNR. Intuitively, when we transmit with high SNR, throughput will definitely increase, however we tend to waste too much transmit power in this way. Meanwhile, if we transmit with low SNR, we suffer from throughput inefficiency. Either way, we might reduce the energy efficiency. Therefore it is critical to choose appropriate transmit SNRs to achieve desirable throughput and energy efficiency under finite delay/rate constraints.

Notice that in (4.8), cumulative transmit energy per bit E_b^{tx} is a function of transmit power E_s and delay constraint D. If we fix D and varies E_s/N_o in 2 dB increments, we are able to find the near-optimal transmit E_s/N_o with the minimum cumulative energy E_b^{tx}. In Fig. 4.10, we plot

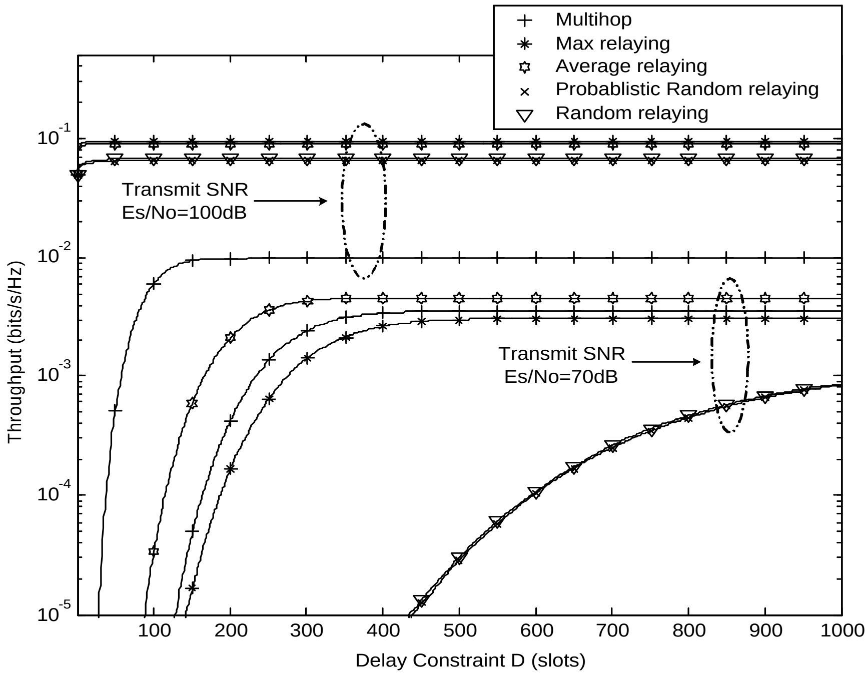

Figure 4.8: The throughput of a network with 10 relays equally spaced along a line as a function of the delay constraint D at both low and high burst transmit SNR. Random relaying denotes interference-free random relaying.

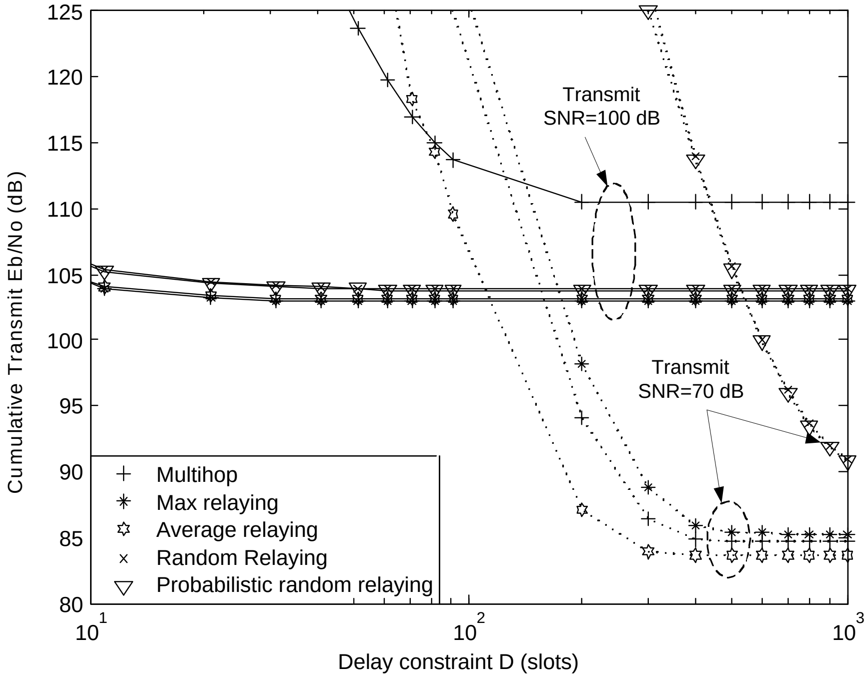

Figure 4.9: The energy efficiency of a network with 10 relays equally spaced along a line as a function of the delay constraint D at both low and high burst transmit SNR. Random relaying denotes interference-free random relaying.

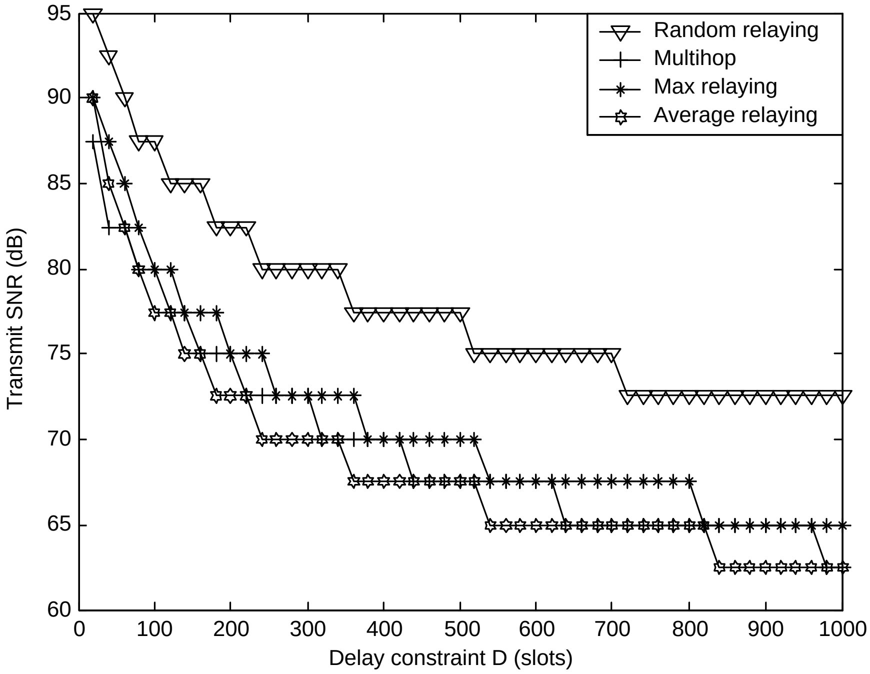

Figure 4.10: The transmit SNR E_s/N_o for each individual protocol to achieve the best transmit-only energy efficiency under finite delay/rate constraint D in a equidistant line network with 10 relays .

out the near-optimal transmit SNR E_s/N_o as a function of delay constraint D for each individual protocol. Applying the same process, we are able to find out optimal transmit SNRs for both direct transmission and a single relay line network. We plot out the energy efficiency under these optimal transmit SNRs in Fig. 4.11. Comparing with Fig.4.2, we notice that when transmitting with near-optimal SNRs, protocols under finite delay/rate constraints perform very close to their performance under infinite delay/rate constraint, especially under loose delay constraints. Therefore, when the delay constraint is large enough, the performance of a specific protocol under finite delay constraint could be closely approximated by its infinite delay counterpart.

The influence of node density on throughput as well as energy efficiency is an important topic for practical system design. In particular, as the node density increases, the available spatial diversity increases accordingly in a relaying network. This will surely benefit the throughput as well as transmit-only energy efficiency. However, more nodes in the network also means that there will be more nodes receiving simultaneously, which will also increase the receive energy dissipation. In Fig. 4.3 and 4.4, we noticed that under infinite delay the dissipated energy efficiency is highly dependent on the ratio of transmitter energy dissipation vs. receiver energy dissipation. In particular, whenever receiver energy dissipation is non negligible compared to transmitter energy dissipation, increasing the node density will reduce the total dissipated energy efficiency, although it does increase transmit-only energy efficiency. The results are straightforward since more and

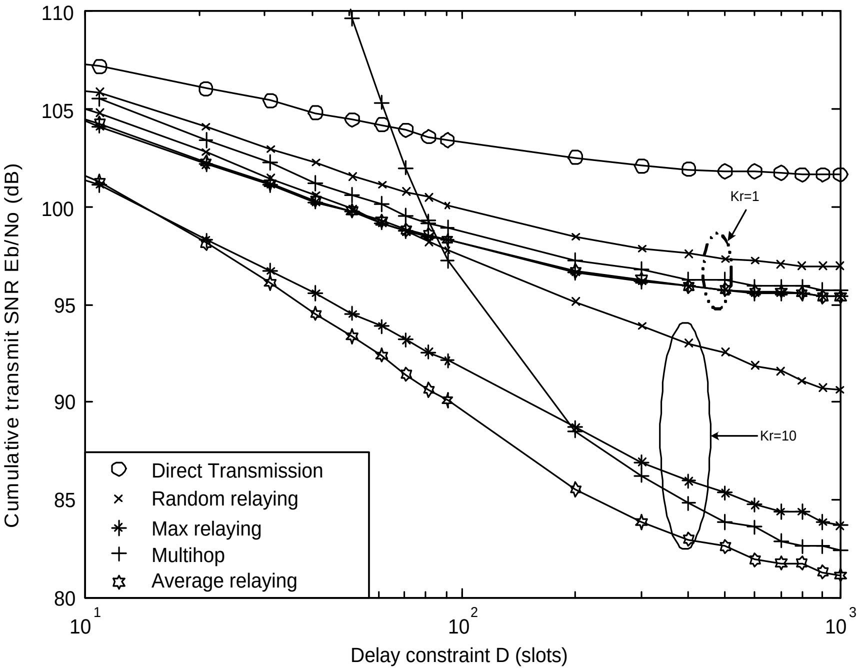

Figure 4.11: The minimum cumulative transmit SNR required to meet delay constraint D in a line network with $K_r = \{1, 10\}$ relays. Results for direct-transmission ($K_r = 0$) are also shown.

more nodes tend to receive the packet thus will waste a lot of receiver energy. However, these results are obtained through infinite delay constraint. In this section, we will further induce the effect of finite delay into this problem and investigate how the dissipated energy efficiency will be affected by node density under different finite delay constraints.

Fig. 4.12 shows the influence of node density on throughput under delay constraint $D = 1000$. For relaying, throughput is a non-decreasing function of the number of relays. With low burst transmit SNR, the system throughput increases dramatically when a few relays are added to the network, although this benefit tends to saturate with a large number of relays. At high SNR, relaying is rather insensitive to the number of relays in the network, since often the initial source transmission will be successful and no relaying will be needed. An interesting observation is that at high SNR, the throughput of multihop decreases almost linearly with respect to the number of relays. This phenomenon is due to the bottleneck effect of multihop's peer-to-peer transmissions.

Fig. 4.13 compares the minimum transmit-only SNR with the minimum total dissipated SNR (including transmit and receive energy dissipation) required to meet the delay constraint $D = 1000$ as a function of the number of relays K_r in an equidistant line network under the assumption that $\mathcal{E}_r = \mathcal{E}_s$. Based on this assumption, we are able to predict the average number of nodes receiving vs the average number of node transmitting. With multihop, each transmitting node has a single corresponding receiving node, thus there is a constant 3 dB difference between the transmit-only

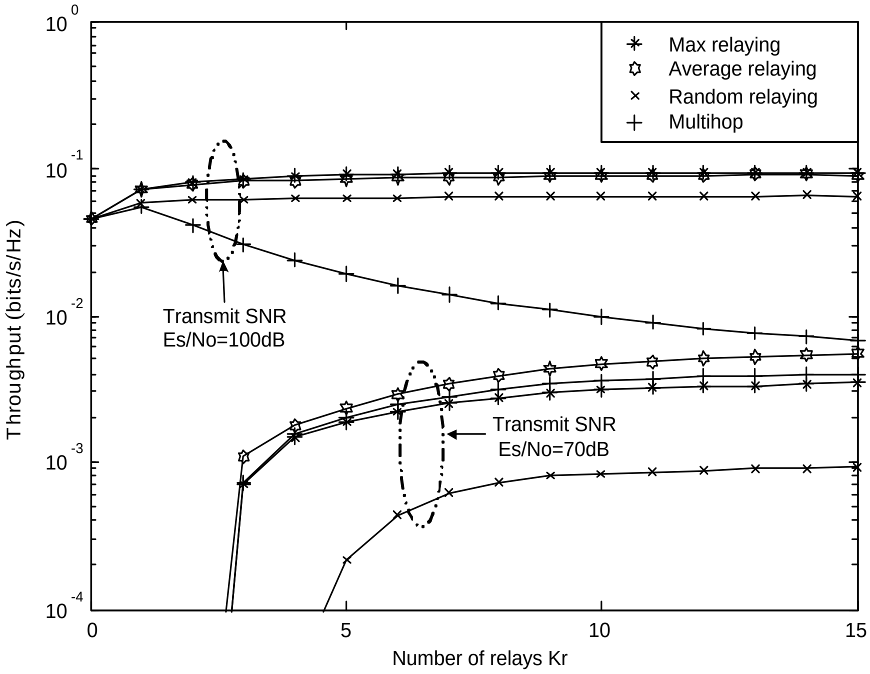

Figure 4.12: The throughput of a line network as a function of the number of relays at low and high burst transmit SNR and delay constraint $D = 1000$.

SNR and the total dissipated SNR. With relaying, this gap increases as more and more relays are added to the network. This is because in relay networks there will usually be many more node receiving the message than broadcasting it. Thus when there are many nodes in the relay network, a significant portion of the total energy is spent receiving the message, not just transmitting it. For instance, when the number of relays is 5, the gap between transmit-only and total dissipated SNR is about $5 \sim 6$ dB, indicating that there are on average 4 nodes receiving each transmission. As the number of relays increases to 15, the gap widens to 10 dB. This means that there are about 9 nodes receiving each transmission. It is the increasing number of receiving nodes that significantly reduce the total dissipated energy efficiency of the relay network.

Assuming that $\mathcal{E}_r = \mathcal{E}_s$ is not realistic in practice, a better comparison is to select a typical value for $\mathcal{E}_r$ and use it for all of the protocols. In the following, we set $\mathcal{E}_r/N_o$ to equal a typical value for $\mathcal{E}_s/N_o$, namely $\mathcal{E}_r/N_o = 80$ dB. Figs. 4.14 and 4.15 show the transmit-only $\mathcal{E}_b^{tx}/N_o$ and total dissipated $\mathcal{E}_b/N_o$ required to meet delay constraints $D = 1000$ and $D = 100$, respectively. The results in Fig. 4.14 indicate that under loose delay constraint, i.e. $D = 1000$, average-relaying always has the best *transmit-only* energy efficiency. However, multihop is most efficient in terms of the total dissipated energy at this delay constraint. The real benefits of relaying are achieved when the delay constraint is rather tight, i.e. for $D = 100$. As shown in Fig.4.15, the transmit-only energy efficiency of each relaying protocol is a monotonically decreasing function of the number of

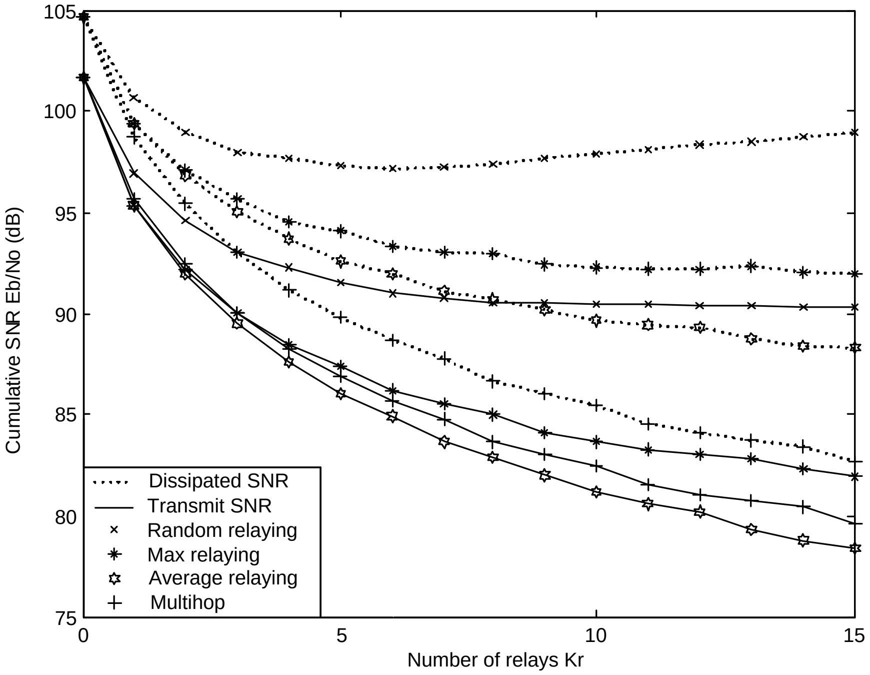

Figure 4.13: The minimum required transmit-only SNR and cumulative SNR for a line network as a function of the number of relays under delay constraint $D = 1000$, assuming that the energy consumed by the circuits that receive a symbol is identical to the energy required to transmit it.

nodes. However, the total dissipated energy efficiency of the relaying protocols begins to rise with an increasing number of relays. The required dissipated energy for multihop also increases with K_r, and this increase is at a much faster rate than that of relaying. Therefore, relaying is much desirable under tight delay constraints.

4.4.2 Probabilistic Transmission: A Practical Design Approach

As we mentioned earlier, interference-free relaying protocols request a highly coordinated global arbitration scheme which is relatively hard to implement in practice. Therefore we consider *probabilistic transmission*, a *distributed* arbitration scheme where each device is assigned a particular transmission probability only related to its own status information, including channel SNR, geographic location, etc. With probabilistic transmission, each device acts autonomously, thus avoiding unnecessary transmission of status information from other devices. Notice that probabilistic transmission does not necessarily avoid collision, however, due to the capture effect, a receiver could still correctly decode even in the presence of interference provided that it receives any one of the transmissions with high enough SINR to make its accumulated mutual information greater than the rate. Therefore explicit collision avoidance mechanisms (e.g. CSMA) are not strictly necessary with this protocol. In this subsection, we will discuss several probabilistic transmission schemes to approximate the interference-free relaying protocols. It is straightforward that interference-free

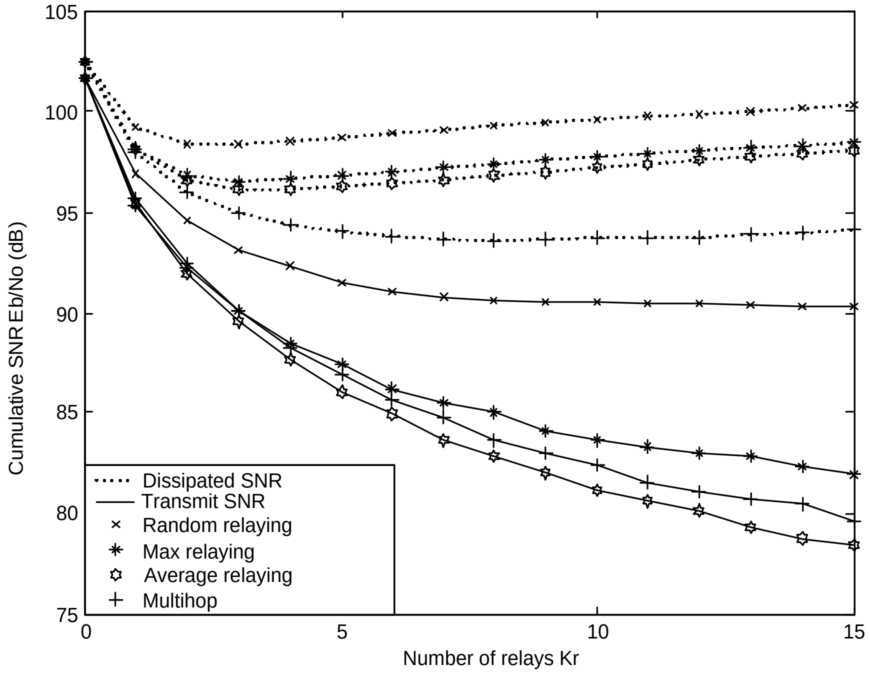

Figure 4.14: The minimum required transmit-only SNR and cumulative SNR for a line network as a function of the number of relays under delay constraint $D = 1000$, assuming that the receive energy dissipation is fixed such that $\mathcal{E}_r/N_o = 80$ dB.

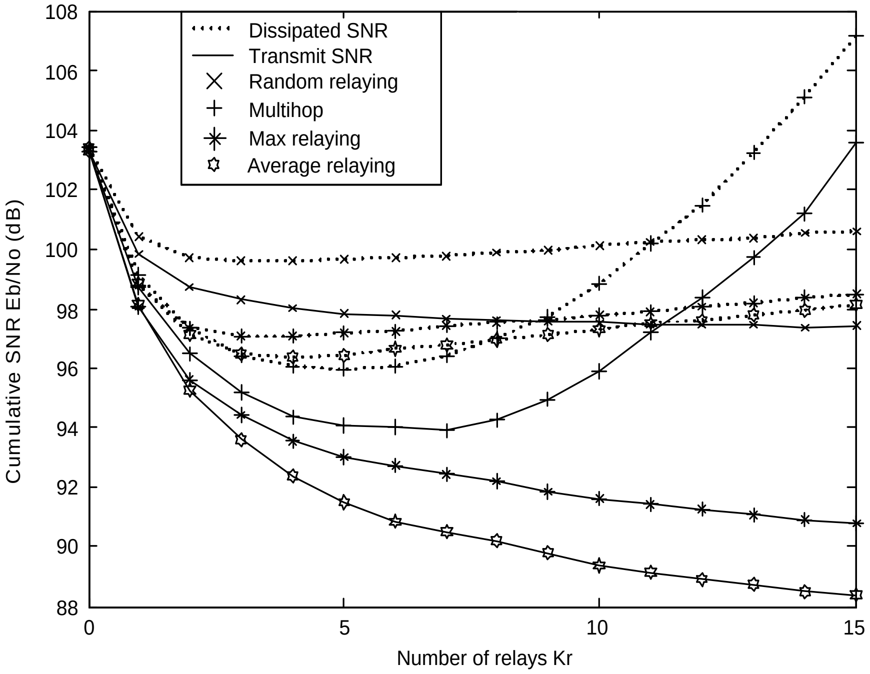

Figure 4.15: The minimum required transmit-only SNR and cumulative SNR for a line network as a function of the number of relays under delay constraint $D = 100$, assuming that the receive energy dissipation is fixed such that $\mathcal{E}_r/N_o = 80$ dB.

relaying serves as an upper bound for their corresponding probabilistic transmission schemes.

Probabilistic Random-Relaying

Consider a probabilistic random-relaying protocol where each node in $\mathcal{D}(s)$ will transmit during slot s with probability $p_k[s]$. Notice that unlike interference-free random relaying where at most one node transmits each time, with probabilistic random-relaying there might be multiple relays transmitting simultaneously. However, since the decision to transmit is autonomous, it is much easier to implement than interference-free random relaying. Because this strategy does not coordinate transmissions, it is possible that multiple nodes simultaneously transmit. However, due to the capture effect, a receiver could still correctly decode even in the presence of interference provided that it receives any one of the transmissions with high enough SINR to make its accumulated mutual information greater than the rate. Therefore explicit collision avoidance mechanisms (e.g. CSMA) are not strictly necessary with this protocol. One major drawback with probabilistic random relaying is that we have virtually no control over the message flow. In random networks, message might spread in an unexpected direction thus might not only cause unnecessary interference to other network clusters but also reduce the energy efficiency of its own cluster.

Fig. 4.8 and 4.9 show the performance comparison between interference-free random relaying and probabilistic random relaying in an equidistant line network with 10 relay nodes under network traffic $p_t = 0.1$. In the simulation, we normalize the node activity by the cardinality of $\mathcal{D}(s)$, e.g. $p_k[s] = 1/|D(s)|$, so that the average network traffic will be the same as with the interference-free relaying protocols, thereby providing a fair comparison. In practice, this normalization would not strictly be necessary, although $\mathcal{D}(s)$ could be estimated by the use of relay-ACKs. Notice that their performances are very close to each other under almost all delay/rate constraints and transmit SNRs. Intuitively when network traffic p_t is small, the collision probability is almost negligible. Therefore, the performance of probabilistic random-relaying could be upper-bounded by that of interference-free random-relaying and the bound is tight under small network traffic p_t. In short, we only need to investigate the interference-free random-relaying protocol, and its results could be readily applied to predict the performance of probabilistic random-relaying.

Probabilistic Average-Relaying

Asymptotic analysis of interference-free protocols indicate that average-relaying generally outperforms max-relaying scheme, and it is much easier to implement. To incorporate probabilistic transmission into average-relaying protocol, we consider probabilistic average-relaying based on the concept of zone splitting [7]. In particular, the broadcast region between the source and the destination is split into N_p priority zones based on the relative distance of each individual zone to the

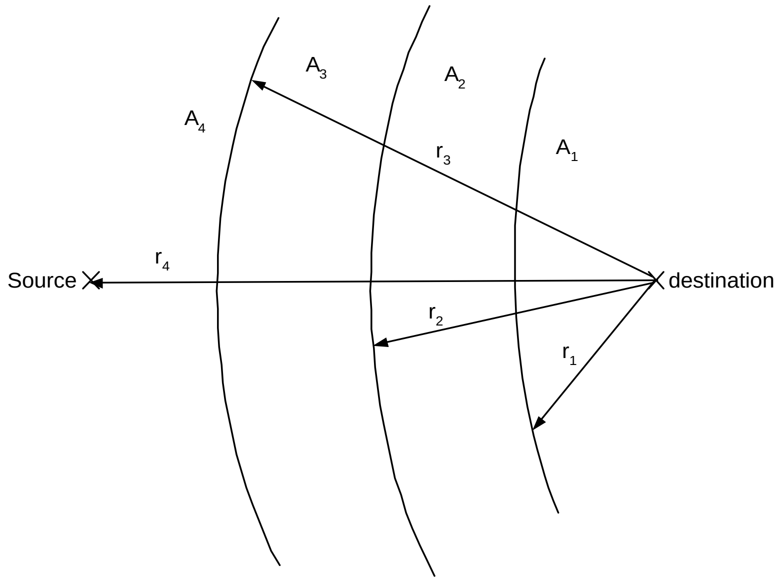

Figure 4.16: A broadcast region between the source and destination is split into 4 priority zones according to their relative distance to the destination.

destination. The zone closer to the destination has higher priority and vice versa. Any decoding node within higher priority zone is more likely to transmit than the decoding node in the lower priority zone. In particular, each node in the broadcast region could calculate its corresponding priority zone based on the following rules

$$\forall Z_i \in D(s) \quad \begin{cases} Z_i \in A_1 & \text{if } |Z_i - Z_d| \le r_1 \\ Z_i \in A_j & \text{if } r_{j-1} \le |Z_i - Z_d| \le r_j,\ 2 \le j \le N_p \end{cases} \tag{4.10}$$

where Z_i also denotes the geographic location of node Z_i. For notational convenience, we also use A_i to denote the set of relay nodes in the priority zone A_i. Fig.4.16 shows an example of zone splitting with $N_p = 4$. The zone splitting is not necessarily uniform, e.g. radius $\{r_i\}$ does not have to increase linearly. More generally, we could provide zones closer to the destination with higher resolution. In general, if the broadcast region is split into more priority zones, it will increase the energy efficiency. However, it will also complicate the implementation. Each priority zone is assigned a particular transmission probability $\left\{p_{A_j} = \frac{(Q-1)}{Q^j}, 1 \le j \le N_p\right\}$. It is straightforward that the assignment will result in an exponential decay on the transmission probability of each priority zone in the broadcast region. $Q > 1$ is a constant chosen to control the speed of decay. For each decoding node to calculate its corresponding priority zone A_i, a message head is attached to each packet, indicating the geographic location of message source and destination. The packet head is encoded with a low rate code such that it could be decoded correctly with high probability. A dedicated relay-ACK channel is used to provide feedback from the relay nodes to the source. Meanwhile, it could also be used by the relay nodes to estimate the decoding set. The relay-ACK

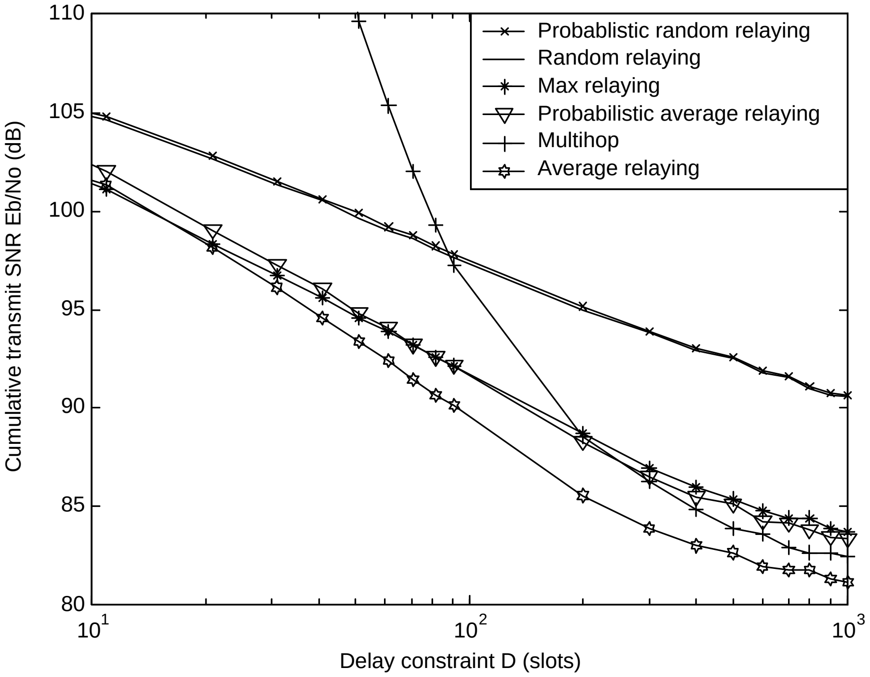

Figure 4.17: The energy efficiency of probabilistic transmission schemes vs. their corresponding interference-free relaying protocols in an equidistant line network with 10 relays. With probabilistic average-relaying, $N_p = 8$ and $Q = 4$.

channel is split into four TDMA sub channels with each channel assigned to the nodes within the corresponding priority zone. Each node recently added to the decoding set will broadcast an ACK message using its corresponding sub channel. In this way, each node in the relay network is able to keep a rough estimate of the decoding set information.

Once each decoding node has figured out its priority zone a corresponding transmission probability is assigned to that node, e.g.

$$p_k = p_t \frac{p_{A_i}}{|D(s) \cap A_i| \sum\limits_{\{\forall j: A_j \neq \phi\}} p_{A_j}} \qquad \forall k \in A_i \tag{4.11}$$

Notice that (4.11) involves a normalization process based on the decoding set profile in order to keep a relatively constant network traffic. Here we do not claim any optimality of this solution, rather we tend to consider it as an efficient means to practically approach the performance of interference-free average relaying protocol.

Fig. 4.17 shows the performance of probabilistic transmission protocols, e.g. probabilistic random relaying and probabilistic average relaying in an equidistant line network with 10 relays. In probabilistic average relaying, we apply uniform zone splitting with $N_p = 8$ and set $Q = 4$, thus the radius r_i increases linearly. The performance of interference-free relaying protocols is also shown for comparison. The network traffic coefficient $p_t = 0.1$ and the number of priority zones is set to be 8. Notice that probabilistic random relaying has exactly the same performance as its interference-free counterpart. However, probabilistic average relaying protocol loses about 2 dB energy efficiency to interference-free average relaying primarily due to the low resolution in zone splitting. As the number of priority zone increases, the performance will approach that of average-relaying. Therefore, based on the observation in Fig. 4.17, we consider probabilistic transmission as an efficient means to approximate interference-free relaying in practical network implementation.

4.5 Summary

In this chapter, we extended our study of the rate constrained relay channel to rate constrained relay networks under larger rate constraint, i.e. $M > 2$. The solution we advocate is to employ hybrid-ARQ to guide the message transmission in the relay networks. In particular, several relaying protocols were proposed and their performance investigated and compared against multihop and direct transmission in terms of throughput, delay, and energy-efficiency. Unlike point-point hybrid-ARQ, the energy consumption in relay networks is shared among all the participating nodes. It was shown that for a network of relays equally spaced along a line, that relaying presents a better tradeoff between delay and energy-efficiency than multihop. The same holds true for a generalized line network consisting of equally spaced clusters of multiple relays, although in the clustering situation the best of relaying protocol depends on the particular configuration.

One drawback of relaying is that many devices must now listen to each broadcast, in contrast with multihop where only a single device receives each transmission. We considered the impact of a non-negligible energy cost to receive a transmission and found that the benefits of relaying begin to diminish when the cost to receive a symbol is on the order of the cost to transmit it. Thus, relaying might be more suitable for transmissions over longer ranges, where transmit power dominates receiver circuit dissipation.

A key advantage of relaying is that it does not require a network-layer protocol to explicitly select a route through the network a priori. Rather, relaying will adaptively find the best "path" and will tend to bypass relays that are continually in an outage. An analogy of this network-layer benefit of ARQ is found at the physical-layer: just as an ARQ-based relay network does not require a network-layer protocol to select the exact route, an ARQ based peer-peer communication system does not need the physical layer to select the optimal code rate. Thus, just as ARQ makes the system robust to incorrect choices of code rate, it also makes the system robust against incorrect route selection. This can be seen in the numerical results by the relative insensitivity of relaying to the number of participating relays. This is in contrast to multihop, where performance can actually degrade if too many relays are selected.

Chapter 5

Hybrid ARQ-Based Intra-cluster Geographically-informed Relaying

5.1 Introduction

In chapter 4, we discussed energy efficient relaying protocols that operate across several network layers within a cluster of cooperating devices. The design and analysis of such protocols was based on the assumption that network topology is constant and all the devices within the network should remain awake and listening to the channel all the time, which consumes considerable processing and transceiver power [74]. This drawback becomes especially obvious for sensor networks that typically must last for a year or more without battery replacement. It is now widely recognized that the most effective way to conserve energy in a sensor network is to periodically put each radio into a sleep mode. The lifetime of the network is dependent on the duty cycle of the nodes, and networks whose nodes are in a sleep state for a higher percentage of time will last longer. However, aggressive power-off strategies impose new challenges in the design of network protocols especially for media access control (MAC) and message routing, since conventional MAC and routing schemes require all nodes to keep listening to the channel to avoid collisions and maintain sufficient network connectivity. Thus, periodically cycling devices into a sleep state will significantly affect MAC and routing efficiency.

Several protocols have been recently proposed for sensor networks with sleeping nodes. A complete survey is outside the scope of this paper, but the interested reader is referred to [7] and the references therein. There are two distinct categories of such protocols: (1) Nodes are awaken according to a deterministic rendezvous schedule, and (2) Nodes cycle on-an-off at random. This chapter focuses on the second type of protocol, as it lends itself to simpler implementation by allowing each node to autonomously set its own sleep schedule.

If nodes know their own position and messages are addressed by location, rather than by MAC address, then it is possible to use this geographic information to guide the routing mechanism. A recently proposed protocol that uses this concept is Geographic Random Forwarding (GeRaF) [7]. With GeRaF, the message is broadcast to all nodes within range and the one node that both decodes the message and is closest to the destination will be the one to forward it. This has the benefit of not requiring a route to be established prior to transmission. Furthermore, it takes advantage of the spatial diversity in the network due to the presence of multiple nodes. In environments with fading and interference, this distributed spatial diversity could allow dramatic improvements in performance if properly exploited [65]. However, a key issue with GeRaF is that the relaying node must be selected in a distributed fashion, without being guided by a genie. This implies that the routing and MAC protocol must be designed jointly. A key contribution of [7] is the development of a practical MAC protocol that allows the most geographically advantaged relay to be chosen after the source has transmitted. This involves a novel process whereby the *receivers* contend for the channel.

While GeRaF is an effective protocol, it tends to require a dense distribution of active nodes. If no node is within range of the source, GeRaF waits until the sleep state changes and then starts over again in the hopes that a node within range has awakened. If the density of active nodes is insufficient, the source may need to wait several sleep/wakeup cycles before there is any forward progress. However, there may be active nodes just out of the source's range that could be used. While these nodes are too far away to successfully decode the initial source transmission, they might be able to decode after the second (or later) transmission if they combine all the information they have received. This is the underlying concept behind *hybrid-ARQ* [21, 58]. With (type-II) hybrid-ARQ, each node will combine all received transmissions prior to decoding a particular message. Based on the study in chapter 3, two alternatives are possible: (1) The message is repeated by the source (repetition-coding); the receiver *diversity-combines* the repeated transmissions, and (2) The source encodes the message by a low rate code and through rate-compatible puncturing a distinct portion of the codeword is transmitted each time (incremental-redundancy); the receiver *code-combines* the received code fragments. With diversity-combining the receiver sees a channel with a higher effective SNR, while with code-combining it receives a code with a lower effective rate. Because the capacity of code-combining is always at least as good as the capacity of diversity-combining [21], that will be the focus of the remainder of this discussion.

Since GeRaF assumes that nodes outside range r_1 don't hear the message, in an AWGN channel and in the absence of interference, only nodes within radius r_1 can be reached during the initial transmission. If there are no geographically advantaged nodes within this range, the source must transmit again. Likewise, during the second transmission, the nodes to be reached are limited to those only within range r_1. However, with the proposed protocol, nodes beyond of range r_1 (say, out to range r_M) receive and maintain each transmitted packet. Thus during the second transmission the *effective* range of the source increases from r_1 to r_2. The effective range continues to increase after each retransmission until a maximum range r_M is reached. As the range increases, so does the probability of finding a geographically advantaged node. Thus, by using this protocol it is possible to use a lower density of nodes than with GeRaF, yet achieve the same delay and consume almost the same (transmit) energy. Because the new protocol uses a combination of GeRaF and hybrid-ARQ, we give it the descriptive name *hybrid ARQ-Based Intra-cluster GEographically-informed Relaying (HARBINGER)*. Notice that once hybrid-ARQ is incorporated into GeRaF, the resulting scheme is very similar to the average relaying protocol discussed in chapter 4, since they both rely on geographic information to forward the message. The major difference is that HARBINGER assumes that nodes outside the coverage area did not receive while average relaying protocol assumes that network devices do not sleep. According to the simulation results in the last chapter, applying hybrid-ARQ to GeRaF has the same advantage as average-relaying, e.g. a better tradeoff between energy efficiency and system throughput compared to multihop routing.

In this chapter, we will develop two versions of HARBINGER with considerably different behavior. The two versions of the protocol differ primarily in the relation between the periodicity of the sleep cycle and the data packet transmission rate. Slow-HARBINGER corresponds to the scenario where nodes cycle in and out of sleep states at a rate that is slower than the data packet rate. Thus, the topology remains fixed for several consecutive packet transmissions, i.e. all retransmissions of the same message, while in Fast-HARBINGER the sleep states are synchronized with the data packet rate. Thus, each time a message is retransmitted, the topology changes. This provides an additional time-diversity benefit relative to Slow-HARBINGER at the expense of requiring more rapid cycling in and out of sleep states.

5.2 Geographic Random Forwarding: A Brief Overview

Based on the system model in chapter 2, we further assume that nodes are randomly distributed according to a Poisson process in a two-dimensional ad hoc wireless network. Due to the use of a periodic sleep cycle, the density ρ of active nodes per unit area is much lower than the actual node density in the area and the topology periodically changes. The *network coherence time* τ is defined to be the amount of time that the topology remains fixed. The first *network coherence interval* (NCI) is defined as the range $t : \{0 \leq t < \tau\}$ while the i^{th} NCI is $t : \{(i-1)\tau \leq t < i\tau\}$. As in [7], we assume each node knows its own position and has a circular coverage area. Nodes within the coverage circle successfully decode the initial transmission, while those outside the circle do not.

For analytical simplicity, we assume an AWGN channel with exponential path-loss and capacity-approaching channel coding (e.g. turbo or LDPC codes) but neglect the influence of interference and fading.

As in GeRaF, there are both message-bearing (data) packets and signalling packets (RTS, CTS, ACK). RTS and CTS packets are primarily used to avoid collision in the network. The signalling packets contain the location of the source and destination and is encoded with a low rate code for maximum error protection so that it could reach all nodes within the coverage area. Once a node has a message to transmit, it will start handshaking at the beginning of the next NCI by sending a RTS signaling packet to detect if there is a potential relay nearby. A node is said to be *geographically advantaged* if it is closer to the destination than the source is, and only geographically advantaged nodes may serve as a relay. If there is no such relay to be found, the source will restart handshaking in the next NCI. Hopefully, through random node activity, a potential relay will appear and respond with a CTS packet indicating that it is ready to receive the subsequent data packet. If multiple potential relays respond with CTS packets, the source will use a contention scheme to choose a particular relay (ideally, the most geographically advantaged). In particular, the relay region of the message source (a region in the coverage circle where all the relay nodes are located) is sliced into N_p priority zones based on the distance from the message sink. Zones closer to the message sink have higher priority. As in GeRaF, nodes in the highest priority zone contend first. If no nodes are found, the nodes in the second highest priority zone contend, and so on until some relay node is found to forward the message.

5.3 The HARBINGER Protocol

With GeRaF if the active node density is fairly low (due to low duty cycle nodes), it is highly probably that the source will need to attempt transmission of the same message in the next NCI, which greatly increases delay. To overcome this drawback, HARBINGER incorporates hybrid Automatic Repeat reQuest (ARQ) into GeRaF. With hybrid-ARQ, distant nodes outside of the source's first attempt transmission range accumulate additional information after each retransmission until they are eventually able to decode. Equivalently, the coverage circle increases after each transmission (since the effective code rate decreases). In HARBINGER, the source will first encode the message with a low rate mother code. The mother codeword is then partitioned into M data packets, where each packet is a distinct portion of the low rate mother code (achieved through rate compatible puncturing). There are essentially two different versions of HARBINGER with respect to the network coherent time τ. In particular, if the network coherence time τ is long enough such that several ARQ retransmissions, i.e. M, can be attempted before the topology changes, then each message could be successfully delivered within a single NCI. This is the underlying assumption for Slow-HARBINGER. Alternatively, if τ is only long enough to accommodate a single ARQ retransmission, then a successful message transmission might span a session of at most M NCIs with a different network topology for each individual NCI. Fast-HARBINGER is designed particularly for such network scenario. Notice that GeRaF is only a special case of HARBINGER with rate constraint M restricted to 1. Since routing and media access control are handled jointly, GeRaF and HARBINGER are both cross-layer protocols.

In general, HARBINGER preserves almost all the packet structure as well as handshaking and contention schemes in GeRaF. In fact, Slow-HARBINGER is a straightforward extension of GeRaF, where a maximum of M data packets could be transmitted after the initial source/relay handshaking at the beginning of each NCI. However, unlike GeRaF, the coverage area of Slow-HARBINGER is expanded to r_M due to the hybrid ARQ-based retransmission scheme. Based on different design criterion, there are two variants of Slow-HARBINGER. On the one hand, we could maximize the message progress within each NCI, resulting in minimum message delay (Slow-HARBINGER A). More specifically, with Slow-HARBINGER A, the source picks the relay node that is closest to the destination to forward the message. On the other hand, we could minimize the number of data packet transmissions per NCI (Slow-HARBINGER B). The reason to minimize data packet retransmission is based on the assumption that data packets are much longer than signalling packets, and thus consumes most transmit energy in the system. In particular, with Slow-HARBINGER B, we pick the relay node that is reachable with minimum number of ARQ retransmissions. However, if multiple nodes require the same number of retransmissions, we will pick the one with the highest priority. Notice that Slow-HARBINGER B does not maximize the message progress, thus will cause extra message delay compared to Slow-HARBINGER A.

If Slow-HARBINGER reduces the message delay through coverage circle expansion, Fast-HARBINGER cuts down the message delay by intentionally causing the topology to change after each ARQ transmission, resulting in a reduction of network coherence time τ. In fact, the delay reduction mechanism of Fast-HARBINGER comes from two aspects, namely the same coverage expansion benefit of Slow-HARBINGER and a time diversity benefit due to shorter τ. Once we assume that data packets are much longer than signalling packets, the sleep schedule in Fast-HARBINGER is essentially synchronized with the data packet transmission rate and thus the network coherence time is very close to the data packet duration.

Fast-HARBINGER is quite different from both GeRaF and Slow-HARBINGER. In particular, Fast-HARBINGER transmits each message via a communication session which spans at most M NCIs. At the beginning of each NCI, the source broadcasts an RTS packet. The RTS packets are numbered 1 through M to indicate which data frame will be sent if a node replies with a CTS packet (the session *expires* after the M^{th} RTS packet is broadcast). All nodes within range $r_M > r_1$ will be able to decode the RTS packet and then must make a local decision to continue listening. This decision is based on its relative location and how many more data packets for the current message could be transmitted. If it is impossible for the node to decode the message before the session expires (i.e. it woke up too late such that it missed too many data packets in the session), then it will go back to sleep. Otherwise, it will send a CTS packet and remain awake for the remainder of the session, even if it is not yet able to decode the message. Upon receiving the first CTS packet, the source will begin to transmit coded packets (one per NCI) until one of the relays is able to decode it and replies with an ACK. Note that if each node's decision to remain awake is perfect, then once a CTS packet is sent the message will always be decoded by *some* relay.

In HARBINGER, once a node decides to receive packets, it will keep every packet it receives so that old information may be combined with fresh information gained after each new ARQ transmission. Eventually this node will be able to decode the message, although it is possible that some other node decodes it first. If multiple nodes successfully decode the message after the same packet transmission, then a contention scheme similar to that in GeRaF could be used to choose the single relay that is most geographically advantaged. Once a specific relay is chosen, all the active nodes within the coverage area will flush their memory (discard previously received packets) and start a brand new session. This memory flushing process is assumed here only to facilitate the analysis. In practice, flushing the memory will reduce performance because the benefit of *distributed* diversity is lost [65].

5.4 A Mathematical Framework

The coverage area of HARBINGER increases after each retransmission due to the ability of hybrid-ARQ to fuse information. Let the effective coverage radius after the m^{th} transmission be r_m. Under the assumption of capacity-approaching channel coding, exponential path loss, and AWGN, this radius can be found by first finding the channel capacity after the m^{th} transmission:

$$C_m = \frac{m}{2}\log_2\left(1 + K_0 d_m^{-\mu}\frac{\mathcal{E}_s}{N_o}\right) \tag{5.1}$$

where $\frac{\mathcal{E}_s}{N_o}$ is the transmit signal to noise ratio, μ the path loss coefficient, K_0 the signal propagation coefficient, d_m the propagation distance. The expression (5.1) follows from the well known result that the capacity of parallel Gaussian channels adds [67]. Any node within the circle of radius d_m is guaranteed to correctly receive the source message of rate $r < C_m$ with no more than m transmissions, where r is the rate of each packet (assumed to be constant for all packets). Solving for the distance d_m we get

$$d_m < \left(\frac{K_0 \mathcal{E}_s / N_o}{2^{2r/m} - 1} \right)^{1/\mu} \tag{5.2}$$

The radius is the maximum transmission range, i.e. $r_m = d_m$. For analytical convenience, we assume $r_1 = 1$ and normalize $\{r_m\}$ with respect to r_1, resulting in

$$r_m = \left(\frac{2^{2r} - 1}{2^{2r/m} - 1} \right)^{1/\mu} \tag{5.3}$$

Note that GeRaF corresponds to the special case that $M = 1$ and thus the normalized transmission radius is always unity. In contrast, the coverage of HARBINGER increases after each transmission.

For notational convenience, we consider the same network setup as [7] where a message source located at $(D, 0)$ while the destination located at $(0, 0)$. The hybrid-ARQ scheme with rate constraint M results in the partitioning of the coverage area by M concentric circles centered at $(D, 0)$. Each circle O_m of radius r_m corresponds to the region where the nodes within could be reached through at most m transmitted packets. Although the actual coverage circle starts at O_1, we consider the point $(D, 0)$ as a virtual circle O_0 of radius $r_0 = 0$ for notational consistency.

Likewise, we could further define $D\nu$ concentric circles centered at $(0, 0)$ with each circle Q_i of radius i/ν . The concentric circles $\{Q_i\}$ quantize the whole range of possible distance from the source to the destination, e.g. $0 \sim D$, into $D\nu$ intervals with ν being the number of quantization interval per unit distance. The l^{th} interval $\triangle_l = (\frac{l}{\nu}, \frac{l-1}{\nu}]$ corresponds to a region $Q_l - Q_{l-1}$. Notice that in HARBINGER, once a relay node is chosen to forward the message, it automatically becomes the source in the next session. Therefore, the distance between source and destination changes each time the message is decoded and the selected relay takes on the role of the source.

Suppose the source is $D = \frac{j}{\nu}$ away from the destination, then partition ξ_m could be defined as

$$\xi_m = (O_m - O_{m-1}) \cap Q_j \quad \text{for } 1 \leq m \leq M \tag{5.4}$$

Any active node in ξ_m could correctly decode the message by receiving exactly m data packets from the source. It is straightforward to show that

$$\cup_{m=1}^{M} \xi_m = O_M \cap Q_j \tag{5.5}$$

$$\xi_i \cap \xi_k = \phi \quad \text{for } i \neq k \tag{5.6}$$

Given $j - r_m \nu + 1 \leq l \leq j - r_{m-1}\nu$, ξ_p is further divided into three disjoint regions by $\triangle_l$ for $\forall p \geq m$. In particular,

$$\xi_p = \xi_{p,1} \cup \xi_{p,2} \cup \xi_{p,3} \tag{5.7}$$

$$\xi_{p,1} = (O_p - O_{p-1}) \cap (Q_j - Q_l) \tag{5.8}$$

$$\xi_{p,2} = (O_p - O_{p-1}) \cap (Q_l - Q_{l-1}) \tag{5.9}$$

$$\xi_{p,3} = (O_p - O_{p-1}) \cap Q_{l-1} \tag{5.10}$$

However, when $\forall p < m$, ξ_p is not further partitioned.

5.5 Performance Analysis

In this section, we characterize the message delay and energy efficiency of both GeRaF and HARBINGER under the same mathematical framework. As we assume that data packets consume most transmit energy in the system, the energy analysis is equivalent to finding the average number of data packet transmission per message.

We notice that with GeRaF or Slow-HARBINGER the number of active nodes in disjoint partitions are independent Poisson random variables at any NCI and the node distribution is time-independent. Thus, in the analysis of GeRaF and Slow-HARBINGER, we use X_i to denote the event that partition ξ_i contains at least one potential relay, while $\bar{X}_i$ denotes the event that partition ξ_i contains no potential relay. Likewise, we use $X_{i,j}$ to denote the event that subpartition $\xi_{i,j}$ contains at least one potential relay, while $\bar{X}_{i,j}$ denotes the event that subpartition $\xi_{i,j}$ contains no potential relay.

However, with Fast-HARBINGER, message transmission spans at most M NCIs, thus the node distribution becomes time-dependent. In particular, when analyzing Fast-HARBINGER, we use X_i^t to denote the event that partition ξ_i contains at least one potential relay at the t^{th} NCI, while $\bar{X}_i^t$ denotes the event that partition ξ_i contains no potential relay during the same time instance. Likewise, we use $X_{i,j}^t$ to denote the event that subpartition $\xi_{i,j}$ contains at least one potential relay at the t^{th} NCI, while $\bar{X}_{i,j}^t$ denotes the event that partition $\xi_{i,j}$ contains no potential relay during the same time instance.

5.5.1 GeRaF

When $M = 1$, HARBINGER reduces to GeRaF. In particular, when the source is within the coverage circle of the final destination, e.g. $D = \frac{j}{\nu}, j = 1, \ldots, \nu$, the message delay is equal to the NCI. On the other hand, when integer $j > \nu$, the random event that the message progress from location $D = j/\nu$ to $\triangle_{j-k+1}$ equals the random event such that there is at least one potential relay

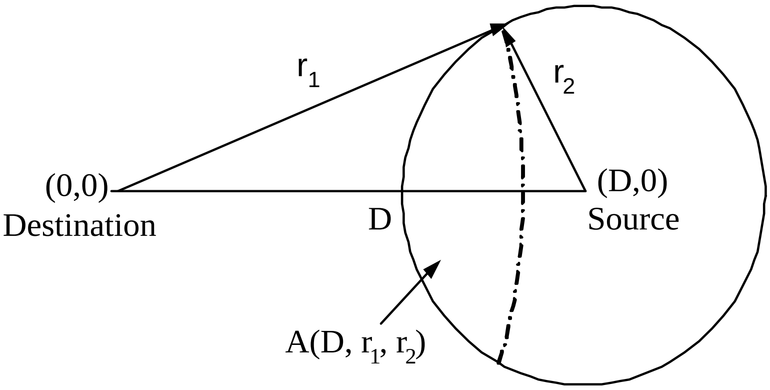

Figure 5.1: The intersection area of two circles of radius r_1 and r_2 separated by a distance of D.

in the partition $\xi_{1,2}$ and no potential relay in partition $\xi_{1,3}$. We use progress probability $\omega(j,k)$ to denote this joint probability,

$$\begin{aligned}\omega(j,k) &= Pr\left\{X_{1,2} \cap \bar{X}_{1,3}\right\} \quad \text{For } k = 1,\dots,\nu \\ &= \exp\left\{-\rho A\left(\frac{j}{\nu}, \frac{j-k}{\nu}, r_1\right)\right\} - \exp\left\{-\rho A\left(\frac{j}{\nu}, \frac{j-k+1}{\nu}, r_1\right)\right\} \quad (5.10)\end{aligned}$$

where

$$A(D,r_1,r_2) = 2\int_{D-r_2}^{r_1} \arccos\left(\frac{x^2+D^2-r_2^2}{2Dx}\right) x\,dx \quad (5.11)$$

denoting the intersection area of two circles of radius r_1 and r_2 separated by a center-to-center distance of D as shown in Fig. 5.1. More specifically, $\omega(j,k)$ denotes the probability of message progress from location $D = j/\nu$ to $\triangle_{j-k+1}$. Both j and k are positive integers.

In addition, ω_0 denotes the probability such that partition ξ_1 contains no potential relay,

$$\begin{aligned}\omega_0(j) &= Pr\left\{\bar{X}_1\right\} \\ &= \exp\left\{-\rho A\left(\frac{j}{\nu}, \frac{j}{\nu}, r_1\right)\right\} \quad (5.11)\end{aligned}$$

We use the same recursive approach in [7] to calculate the upper and lower bounds of average message delay. Further notice that delay is counted as an integer multiple of the NCI, thus in the rest of this chapter we normalize the delay metric with τ for notational elegance. Accordingly, the upper bound $n_1(j)$ and lower bound $n_2(j)$ become

$$n_1(j) = 1 + \omega_0(j)n_1(j) + \sum_{k=1}^{r_1\nu} \omega(j,k)n_1(j-k+1) \quad (5.12)$$

$$n_2(j) = 1 + \omega_0(j)n_2(j) + \sum_{k=1}^{r_1\nu} \omega(j,k)n_2(j-k) \quad (5.13)$$

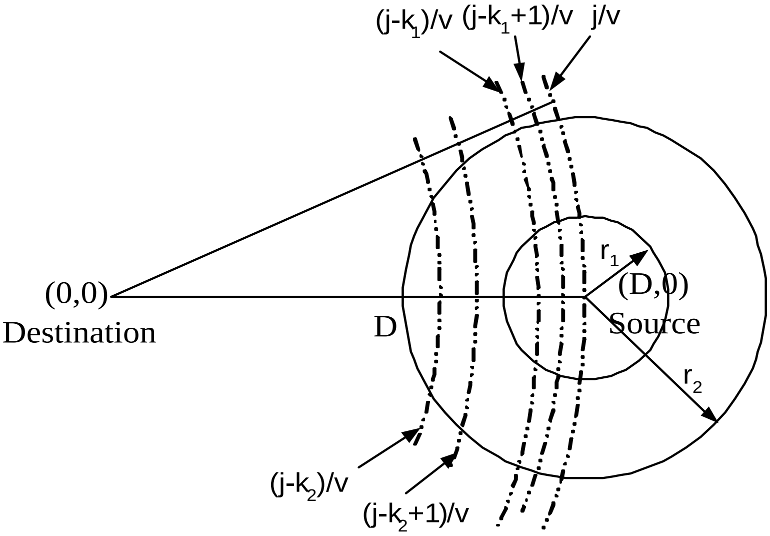

Figure 5.2: Concentric coverage circles for HARBINGER with $M = 2$.

with initial condition $n_1(j) = n_2(j) = 1$ for $j = 1, \dots, \nu$.

Further notice that GeRaF will not transmit data packet when there is no potential relay in the coverage area. Therefore, upper and lower bounds on the average number of data packets per message could be recursively calculated by slightly modifying (5.12)(5.13). More specifically, when integer $j > \nu$,

$$e_1(j) \quad = \quad \omega_0(j)e_1(j) + \sum_{k=1}^{r_1\nu} \omega(j,k)(e_1(j-k+1)+1) \tag{5.14}$$

$$e_2(j) \quad = \quad \omega_0(j)e_2(j) + \sum_{k=1}^{r_1\nu} \omega(j,k)(e_2(j-k)+1) \tag{5.15}$$

The initial condition is set as $e_1(j) = e_2(j) = 1$ for $j = 1, \dots, \nu$.

As $M > 1$, hybrid-ARQ scheme needs to be incorporated into the analysis of HARBINGER. We will first analyze the performance of Slow-HARBINGER and later switch to Fast-HARBINGER.

5.5.2 Slow-HARBINGER A

Notice that Slow-HARBINGER A tends to maximize the message progress within each NCI, therefore the source will always pick the relay node closest to the destination. Consider a simple example of $M = 2$ as shown in Fig. 5.2. When integer $j > r_2\nu$, the destination is outside coverage area O_2. The corresponding probability of message progress $\omega(j,k)$ is found as

$$\omega(j,k) = \begin{cases} Pr\left\{\left(\cap_{i=1}^{2} \bar{X}_{i,3}\right) \cap \left(\cup_{i=1}^{2} X_{i,2}\right)\right\} & \text{for } k = 1, \dots, \nu \\ Pr\left\{\bar{X}_{2,3} \cap X_{2,2}\right\} & \text{for } k = \nu+1, \dots, r_2\nu \end{cases} \tag{5.16}$$

where $\omega(j,k)$ could be further decomposed into

$$\omega(j,k) = \omega(j,k,1,1) + \omega(j,k,1,2) \tag{5.17}$$

where $\omega(j,k,b,l)$ denotes the joint probability of message progress from location j/ν to $\triangle_{j-k+1}$ with exactly $b\tau$ delay and l data packet transmissions. Notice that in Slow-HARBINGER, message transmission is limited to a single NCI, therefore delay $b \equiv 1$. In particular,

$$\omega(j,k,1,1) \;=\; \begin{cases} Pr\left\{\left(\cap_{i=1}^{2}\bar{X}_{i,3}\right) \cap X_{1,2}\right\}, & \text{for } k = 1,\ldots,\nu \\ 0, & \text{for } k = \nu+1,\ldots,r_2\nu \end{cases} \tag{5.18}$$

and

$$\omega(j,k,1,2) \;=\; \begin{cases} Pr\left\{\left(\cap_{i=1}^{2}\bar{X}_{i,3}\right) \cap \bar{X}_{1,2} \cap X_{2,2}\right\}, & \text{for } k = 1,\ldots,\nu \\ Pr\left\{\bar{X}_{2,3} \cap X_{2,2}\right\}, & \text{for } k = \nu+1,\ldots,r_2\nu \end{cases} \tag{5.19}$$

In addition,

$$w_0(j) \;=\; Pr\left\{\cap_{i=1}^{2}\bar{X}_i\right\} \tag{5.20}$$

More generally, as $M > 1$, we have the following message progress probability. On the one hand, when integer $j > r_M\nu$, the destination is outside coverage area O_M. When $k = r_{p-1}\nu+1,\ldots,r_p\nu$, the corresponding message progress probability $\omega(j,k)$ is found as

$$\begin{aligned} \omega(j,k) \;&=\; Pr\left\{\left(\cap_{i=p}^{M}\bar{X}_{i,3}\right) \cap \left(\cup_{i=p}^{M}X_{i,2}\right)\right\} \\ &=\; \exp\left\{-\rho A\left(j/\nu,(j-k)/\nu,r_M\right)\right\} - \exp\left\{-\rho A\left(j/\nu,(j-k+1)/\nu,r_M\right)\right\} \end{aligned} \tag{5.20}$$

where $\omega(j,k)$ could be further decomposed into

$$\omega(j,k) = \sum_{m=p}^{M} \omega(j,k,1,m) \tag{5.21}$$

where $\omega(j,k,b,l)$ denotes the joint probability of message progress from location j/ν to $\triangle_{j-k+1}$ with exactly $b\tau$ delay and l data packet transmissions. Notice that in Slow-HARBINGER, message transmission is limited to a single NCI, therefore delay $b \equiv 1$. In particular,

$$\omega(j,k,1,m) \;=\; \begin{cases} Pr\left\{\left(\cap_{i=p}^{M}\bar{X}_{i,3}\right)\right\} Pr\left\{\cup_{i=p}^{m}X_{i,2} - \cup_{i=p}^{m-1}X_{i,2}\right\} & \text{for } m = p,\ldots,M \\ 0, & \text{for } m < p \end{cases} \tag{5.22}$$

and

$$Pr\left\{\cup_{i=p}^{l}X_{i,2}\right\} = 1 - \exp\left\{-\rho\left(A\left(j/\nu,(j-k+1)/\nu,r_l\right) - A\left(j/\nu,(j-k)/\nu,r_l\right)\right)\right\} \tag{5.23}$$

$$Pr\left\{\left(\cap_{i=p}^{M}\bar{X}_{i,3}\right)\right\} = \exp\left\{-\rho A\left(j/\nu, (j-k)/\nu, r_M\right)\right\} \tag{5.24}$$

In addition,

$$\begin{aligned} w_0(j) &= Pr\left\{\cap_{i=1}^{M}\bar{X}_i\right\} \\ &= \exp\left\{-\rho A\left(j/\nu, j/\nu, r_M\right)\right\} \end{aligned} \tag{5.24}$$

Correspondingly, the upper and lower bounds of average delay become

$$n_1(j) = \sum_{k=1}^{r_M\nu}\omega(j,k)(n_1(j-k+1)+1)+\omega_0(j)(n_1(j)+1) \tag{5.25}$$

$$n_2(j) = \sum_{k=1}^{r_M\nu}\omega(j,k)(n_2(j-k)+1)+\omega_0(j)(n_2(j)+1) \tag{5.26}$$

with initial condition $n_1(j) = n_2(j) = 1$ when $j = 1, \ldots, r_M\nu$. Likewise the average number of data packet transmissions becomes

$$e_1(j) = \sum_{k=1}^{r_M\nu}\sum_{m=1}^{M}\omega(j,k,1,m)(e_1(j-k+1)+m)+\omega_0(j)e_1(j) \tag{5.27}$$

$$e_2(j) = \sum_{k=1}^{r_M\nu}\sum_{m=1}^{M}\omega(j,k,1,m)(e_2(j-k)+m)+\omega_0(j)e_2(j) \tag{5.28}$$

with initial condition $e_1(j) = e_2(j) = m$ when $j = r_{m-1}\nu+1, \ldots, r_m\nu$.

5.5.3 Slow-HARBINGER B

Unlike Slow-HARBINGER A, Slow-HARBINGER B intends to minimize the data packet transmissions per NCI, thus it will increase the message delay. In particular, Slow-HARBINGER B picks the relay node that is reachable with minimum number of ARQ retransmissions. Consider $M = 2$. The progress probability $\omega(j,k)$ is

$$\omega(j,k) = \omega(j,k,1,1) + \omega(j,k,1,2) \tag{5.29}$$

where

$$\omega(j,k,1,1) = \begin{cases} Pr\left\{\bar{X}_{1,3}\cap X_{1,2}\right\}, & \text{for } k = 1, \ldots, \nu \\ 0, & \text{for } k = \nu+1, \ldots, r_2\nu \end{cases} \tag{5.30}$$

$$\omega(j,k,1,2) = Pr\left\{\bar{X}_1\cap\bar{X}_{2,3}\cap X_{2,2}\right\}, \quad \text{for } k = 1, \ldots, r_2\nu \tag{5.31}$$

In addition,

$$w_0(j) \;=\; Pr\left\{\cap_{i=1}^{2}\bar{X}_i\right\} \tag{5.32}$$

More generally, as $M > 1$, we found the following progress probability. On the one hand, when integer $j > r_M\nu$, the destination is outside coverage area O_M. When $k = r_{p-1}\nu + 1, \ldots, r_p\nu$, its corresponding probability $\omega(j, k)$ is found as

$$\omega(j,k) \;=\; \sum_{m=1}^{M}\omega(j,k,1,m) \tag{5.33}$$

where $\omega(j, k, b, l)$ denotes the joint probability of message progress from location j/ν to $\triangle_{j-k+1}$ with exactly $b\tau$ delay and l data packet transmissions. Notice that in Slow-HARBINGER, message transmission is limited to a single NCI, therefore delay $b \equiv 1$.

$$\omega(j,k,1,m) \;=\; \begin{cases} Pr\left\{X_{m,2}\cap\bar{X}_{m,3}\cap\left(\cap_{i=1}^{m-1}\bar{X}_i\right)\right\}, & \text{for } m = p,\ldots,M \\ 0, & \text{for } m < p \end{cases} \tag{5.34}$$

where

$$\begin{aligned} Pr\left\{X_{m,2}\cap\bar{X}_{m,3}\right\} \;=\;& \exp\left\{-\rho\left(A\left(\frac{j}{\nu},\frac{j-k}{\nu},r_m\right)-A\left(\frac{j}{\nu},\frac{j-k}{\nu},r_{m-1}\right)\right)\right\} \\ & -\exp\left\{-\rho\left(A\left(\frac{j}{\nu},\frac{j-k+1}{\nu},r_m\right)-A\left(\frac{j}{\nu},\frac{j-k+1}{\nu},r_{m-1}\right)\right)\right\} \end{aligned} \tag{5.33}$$

and

$$Pr\left\{\cap_{i=1}^{m-1}\bar{X}_i\right\} = \exp\left\{-\rho A\left(j/\nu, j/\nu, r_{m-1}\right)\right\} \tag{5.34}$$

On the other hand, when $j = r_{p-1}\nu + 1, \ldots, r_p\nu$, the destination is within partition ξ_p, thus is reachable with at most p data packets. In particular,

$$w_0(j) \;=\; 0 \tag{5.35}$$

$$\omega(j,k,1,m) \;=\; \begin{cases} Pr\left\{\cap_{i=1}^{p-1}\bar{X}_i\right\}\delta(m,p)\delta(j,k), & k = r_{p-1}\nu+1,\ldots,j \\ Pr\left\{X_{m,2}\cap\bar{X}_{m,3}\cap\left(\cap_{i=1}^{m-1}\bar{X}_i\right)\right\}, & k = r_{l-1}\nu+1,\ldots,r_l\nu \text{ and } m = l,\ldots,p-1 \\ 0, & \text{otherwise} \end{cases} \tag{5.35}$$

Correspondingly, the upper and lower bounds of average delay as well as average data packets could be calculated recursively by (5.25)(5.26) and (5.27)(5.28) with initial condition $n_1(j) = n_2(j) = 1$ when $j = 1, \ldots, \nu$, $e_1(j) = e_2(j) = 1$ when $j = 1, \ldots, \nu$.

5.5.4 Fast-HARBINGER

We have used a recursive approach to characterize the message delay as well as the number of data packet transmissions per message in Slow-HARBINGER. In this section, we will analyze Fast-HARBINGER. Notice that as $M > 1$, the time varying nature of the network topology starts to kick in. In particular, let us first consider the $M = 2$ case where the source session spans at most 2τ.

When $M > 1$ and integer $j > r_2\nu$, we use $\omega(j,k,b)$ to denote the joint probability of message progress from location j/ν to $\triangle_{j-k+1}$ with exactly $b\tau$ delay ($b \leq M$). In particular,

$$\omega(j,k,1) = \begin{cases} Pr\left\{X_{1,2}^1 \cap \bar{X}_{1,3}^1\right\}, & \text{for } k = 1,\ldots,r_1\nu \\ 0, & \text{for } k = r_1\nu+1,\ldots,r_2\nu \end{cases} \tag{5.36}$$

$$\omega(j,k,2) = \begin{cases} Pr\left\{\bar{X}_1^1 \cap \bar{X}_{2,3}^1 \cap \bar{X}_{1,3}^2 \cap \left(X_{1,2}^2 \cup X_{2,2}^1\right)\right\}, & \text{for } k = 1,\ldots,r_1\nu \\ Pr\left\{\bar{X}_1^1 \cap X_{2,2}^1 \cap \bar{X}_{2,3}^1\right\}, & \text{for } k = r_1\nu+1,\ldots,r_2\nu \end{cases} \tag{5.37}$$

where $\bar{X}_1^1 \cap \bar{X}_{2,3}^1 \cap \bar{X}_{1,3}^2 \cap \left(X_{1,2}^2 \cup X_{2,2}^1\right)$ denotes the random event that message progress to $\xi_{1,2} \cup \xi_{2,2}$ with exactly 2τ delay in the session. In addition,

$$\omega_0(j) = Pr\left\{\bar{X}_1^1 \cap \bar{X}_2^1 \cap \bar{X}_1^2\right\} \tag{5.38}$$

denotes the probability that no progress is made within the current session, and thus the same message has to be retransmitted in the next session.

More generally, as $M > 1$, we found the following progress probability. On the one hand, when integer $j > r_M\nu$, the destination is outside coverage area O_M. When $k = r_{p-1}\nu+1,\ldots,r_p\nu$, the corresponding joint probability of message progress $\omega(j,k,b)$ is found as

$$\omega(j,k,b) = \begin{cases} Pr\left\{\Omega\left(\cup_{i=p}^b \xi_{i,2}, b\right)\right\} & \text{for } b = p,\ldots,M \\ 0 & \text{otherwise} \end{cases} \tag{5.39}$$

where

$$\Omega\left(\cup_{i=p}^b \xi_{i,2}, b\right) = \left(\cap_{i=1}^{b-1} \cap_{t=1}^{b-i} \bar{X}_i^t\right) \cap \left(\cap_{i=p}^b \bar{X}_{i,3}^{b+1-i}\right) \cap \left(\cup_{i=p}^b X_{i,2}^{b+1-i}\right) \tag{5.40}$$

$\Omega(A,b)$ denotes the random event that the message advance to the region A with exactly $b\tau$ delay in the session. The probability of these independent Poisson random events could be found as

$$Pr\left\{\cap_{i=1}^{b-1} \cap_{t=1}^{b-i} \bar{X}_i^t\right\} = \prod_{i=1}^{b-1} \exp\left\{-\rho A(j/\nu, j/\nu, r_i)\right\} \tag{5.41}$$

$$\begin{aligned} Pr\left\{\left(\cap_{i=p}^{b}\bar{X}_{i,3}^{b+1-i}\right)\cap\left(\cup_{i=p}^{b}X_{i,2}^{b+1-i}\right)\right\} &= \exp\left\{-\rho A\left(j/\nu,(j-k)/\nu,r_b\right)\right\} \\ &\quad -\exp\left\{-\rho A\left(j/\nu,(j-k+1)/\nu,r_b\right)\right\} \end{aligned} \tag{5.41}$$

In addition,

$$\begin{aligned} \omega_0(j) &= Pr\left\{\cap_{i=1}^{M}\cap_{t=1}^{M+1-i}\bar{X}_i^t\right\} \\ &= \prod_{i=1}^{M}\exp\left\{-\rho A(j/\nu,j/\nu,r_i)\right\} \end{aligned} \tag{5.41}$$

On the other hand, when $j = r_{m-1}\nu+1,\ldots,r_m\nu$, the destination is within partition ξ_m, thus is reachable with at most $m\tau$ delay. However, the analysis becomes different due to the fact that there is always a node, e.g. the destination, within the coverage area. In particular, we have the following results,

$$w_0(j) = 0 \tag{5.42}$$

When $k = r_{p-1}\nu+1,\ldots,r_p\nu$, and $p < m$

$$\omega(j,k,b) = \begin{cases} Pr\left\{\Omega\left(\cup_{i=p}^{b}\xi_{i,2},b\right)\right\}, & \text{for } b = p,\ldots,m-1 \\ 0, & \text{otherwise} \end{cases} \tag{5.43}$$

When $k = r_{m-1}\nu+1,\ldots,j$,

$$\omega(j,k,b) = Pr\left\{\cap_{l=1}^{b-1}\cap_{t=1}^{b-l}\bar{X}_l^t\right\}\delta(j,k)\delta(m,b) \tag{5.44}$$

Correspondingly, the upper and lower bounds of average delay can be generalized as

$$n_1(j) = \sum_{k=1}^{r_M\nu}\sum_{b=1}^{M}\omega(j,k,b)(n_1(j-k+1)+b)+\omega_0(j)(n_1(j)+M) \tag{5.45}$$

$$n_2(j) = \sum_{k=1}^{r_M\nu}\sum_{b=1}^{M}\omega(j,k,b)(n_2(j-k)+b)+\omega_0(j)(n_2(j)+M) \tag{5.46}$$

with initial condition $n_1(j) = n_2(j) = 1$ when $j = 1,\ldots,\nu$ and $n_1(j) = n_2(j) = 2$ when $j = \nu+1,\ldots,r_2\nu$.

To calculate the average number of data packets per message, we use a joint probability $\omega(j,k,b,l)$ to denote the probability of message progress from location j/ν to $\triangle_{j-k+1}$ with exactly $b\tau$ delay and l data packet transmission ($l \leq b$). It is straightforward that

$$\omega(j,k,b) = \sum_{l=1}^{b}\omega(j,k,b,l) \tag{5.47}$$

For simple system as $M = 2$, $\omega(j,k,b,l)$ could be found directly from (5.36) (5.37)as

$$\omega(j,k,1,1) \quad = \quad \omega(j,k,1), \qquad \text{for } k = 1,\dots,r_2\nu \tag{5.48}$$

$$\omega(j,k,2,1) \quad = \quad \begin{cases} Pr\left\{\Omega_2\right\}, & \text{for } k = 1,\dots,r_1\nu \\ 0, & \text{for } k = r_1\nu+1,\dots,r_2\nu \end{cases} \tag{5.49}$$

$$\omega(j,k,2,2) \quad = \quad \begin{cases} Pr\left\{\Omega_1 - \Omega_2\right\}, & \text{for } k = 1,\dots,r_1\nu \\ Pr\left\{\bar{X}_1^1 \cap X_{2,2}^1 \cap \bar{X}_{2,3}^1\right\}, & \text{for } k = r_1\nu+1,\dots,r_2\nu \end{cases} \tag{5.50}$$

where

$$\Omega_1 \quad = \quad \bar{X}_1^1 \cap \bar{X}_{2,3}^1 \cap \bar{X}_{1,3}^2 \cap \left(X_{1,2}^2 \cup X_{2,2}^1\right) \tag{5.51}$$

$$\Omega_2 \quad = \quad \bar{X}_1^1 \cap \bar{X}_2^1 \cap X_{1,2}^2 \bar{X}_{1,3}^2 \tag{5.52}$$

More generally, as $M > 2$, $\omega(j,k,b,l)$ gets fairly complicated. When integer $j > r_M\nu$, and $k = r_{p-1}\nu+1,\dots,r_p\nu$, the joint probability $\omega(j,k,b,l)$ is found as

$$\omega(j,k,b,l) = \begin{cases} Pr\left\{\Omega\left(\cup_{i=p}^{b}\xi_{i,2},b,l\right) - \Omega\left(\cup_{i=p}^{b}\xi_{i,2},b,l-1\right)\right\}, & \text{for } b = p,\dots,M \text{ and } l = p,\dots,b \\ 0, & \text{otherwise} \end{cases} \tag{5.53}$$

where

$$\Omega\left(\cup_{i=p}^{b}\xi_{i,2},b,l\right) \quad = \quad \left(\cap_{t=1}^{b-l}\cap_{i=1}^{M}\bar{X}_i^t\right) \cap \left(\cap_{t=b+1-l}^{b-1}\cap_{i=1}^{b-t}\bar{X}_i^t\right) \cap \left(\cap_{i=p}^{l}\bar{X}_{i,3}^{b+1-i}\right) \cap \left(\cup_{i=p}^{l}X_{i,2}^{b+1-i}\right) \tag{5.53}$$

$\Omega(A,b,l)$ denotes the joint event that the message advance to the region A with exactly $b\tau$ delay and at most l data packets in the session. It is straightforward that

$$\Omega\left(\cup_{i=p}^{b}\xi_{i,2},b,b\right) \quad = \quad \Omega\left(\cup_{i=p}^{b}\xi_{i,2},b\right) \tag{5.54}$$

The probability of these independent poisson random events could be found as

$$Pr\left\{\cap_{t=1}^{b-l}\cap_{i=1}^{M}\bar{X}_i^t\right\} = \left(\exp\left\{-\rho A(j/\nu,j/\nu,r_M)\right\}\right)^{b-l} \tag{5.55}$$

$$Pr\left\{\cap_{t=b+1-l}^{b-1}\cap_{i=1}^{b-t}\bar{X}_i^t\right\} = \prod_{i=1}^{l-1}\exp\left\{-\rho A(j/\nu,j/\nu,r_i)\right\} \tag{5.56}$$

and

$$Pr\left\{\left(\cap_{i=p}^{l}\bar{X}_{i,3}^{b+1-i}\right)\cap\left(\cup_{i=p}^{l}X_{i,2}^{b+1-i}\right)\right\} = \exp\left\{-\rho A\left(j/\nu,(j-k)/\nu,r_l\right)\right\} - \exp\left\{-\rho A\left(j/\nu,(j-k+1)/\nu,r_l\right)\right\} \quad (5.56)$$

On the other hand, when $j = r_{m-1}\nu + 1, \dots, r_m\nu$, the destination is within partition ξ_m, thus the source will always transmit data packet in each NCI. Therefore, $\omega(j,k,b,l)$ becomes

$$\omega(j,k,b,l) = \omega(j,k,b)\delta(b,l) \quad (5.57)$$

Correspondingly, the upper and lower bounds of average number of data packets per message could be calculated recursively through the following equations

$$e_1(j) = \sum_{k=1}^{r_M\nu}\sum_{b=1}^{M}\sum_{l=1}^{b}\omega(j,k,b,l)(e_1(j-k+1)+l)+\omega_0(j)e_1(j) \quad (5.58)$$

$$e_2(j) = \sum_{k=1}^{r_M\nu}\sum_{b=1}^{M}\sum_{l=1}^{b}\omega(j,k,b,l)(e_2(j-k)+l)+\omega_0(j)e_2(j) \quad (5.59)$$

with initial condition $e_1(j) = e_2(j) = 1$ when $j = 1, \dots, \nu$ and $e_1(j) = e_2(j) = 2$ when $j = \nu + 1, \dots, r_2\nu$.

5.6 Numerical Results

5.6.1 Message Delay

The upper and lower bounds of message delay in HARBINGER are plotted in Fig. 5.3, Fig. 5.4, and Fig. 5.6 for Slow-HARBINGER A, Slow-HARBINGER B, and Fast-HARBINGER respectively. The bounds are calculated for different rate constraints, e.g. $M = 2, 12$. The delay performance of GeRaF is also included as a general framework of reference ($M = 1$). In all three figures, the message delay is normalized by their corresponding network (topology) coherence time τ. The block code rate $R = 1$ and the number of interval per unit distance $\nu = 50$. If we change code rate R, the radius $\{r_i\}$ of coverage circles change accordingly which will affect the delay performance. As observed in all three figures, the upper and lower bounds are quite close to each other indicating the tightness of both bounds. In fact, as the interval $1/\nu$ becomes smaller, the bounds will become more accurate and vice versa. Notice that with $M = 1$, both Fast-HARBINGER and Slow-HARBINGER are reduced to GeRaF. In Fig 5.3, we observe that Slow-HARBINGER A significantly reduces the message delay as the rate constraint increases. The result is rather intuitive, since from the

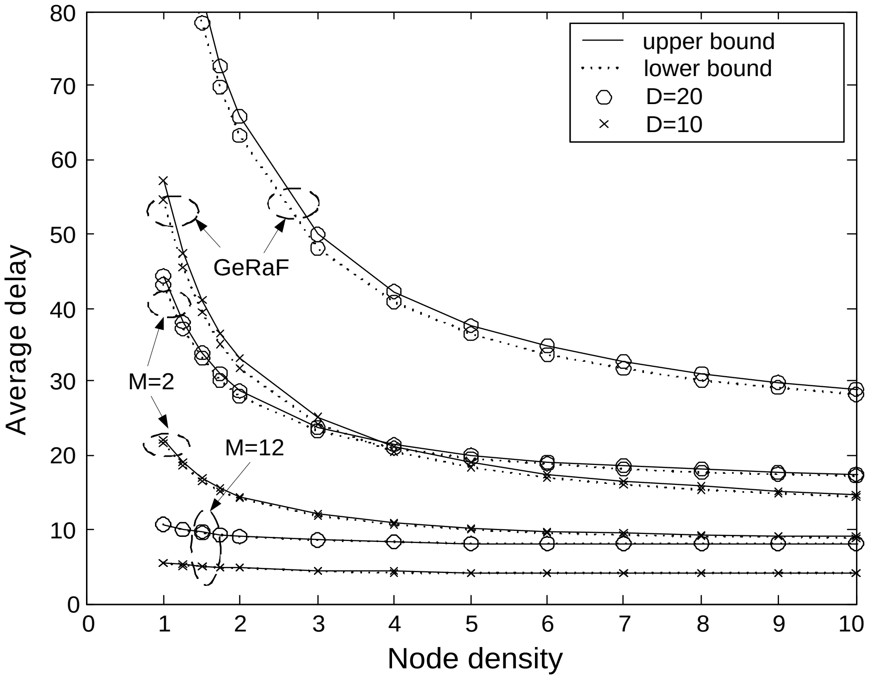

Figure 5.3: The average delay (normalized by τ) of Slow-HARBINGER A under different rate constraints M, $M = 1$ corresponds to GeRaF.

message delay perspective Slow-HARBINGER A is almost equivalent to GeRaF with its coverage circle expanded to r_M. Therefore, asymptotically as active node density $\rho \to \infty$, the message delay will converge to $\lfloor \frac{D}{r_M} + 1 \rfloor$, where D is source/destination separation. Compared with Slow-HARBINGER A, Slow-HARBINGER B has different delay characteristics. As noticed in Fig. 5.4, in a relatively dense network, Slow-HARBINGER B has the same performance as GeRaF. In fact, both GeRaF and Slow-HARBINGER B will asymptotically converge to a message delay of $\lfloor D+1 \rfloor$ as node density $\rho \to \infty$. Note that the performance difference between Slow-HARBINGER A and Slow-HARBINGER B in dense networks is primarily due to their different relay selection criterion. In particular, Slow-HARBINGER A picks the relay closest to the destination while Slow-HARBINGER B picks the relay requiring minimum ARQ retransmissions.

An interesting phenomenon we observed in Fig.5.4 is that as the rate constraint gets fairly large, i.e. $M = 12$, the delay performance is not a monotonically decreasing function of node density. In particular, in low density network, message delay actually decreases along with the node density. This observation is counter-intuitive. To explain this phenomenon, we calculate and plot the average message progress $Avg(j)$ per NCI under different node density in Fig. 5.5, where

$$Avg(j) = \sum_{k=1}^{r_M \nu} \omega(j,k) \frac{k}{\nu} \tag{5.60}$$

$\omega(j,k)$ as a function of node density could be calculated through (5.33). Notice that in Fig. 5.5 the message progress is actually larger in networks with lower density, indicating that nodes closer

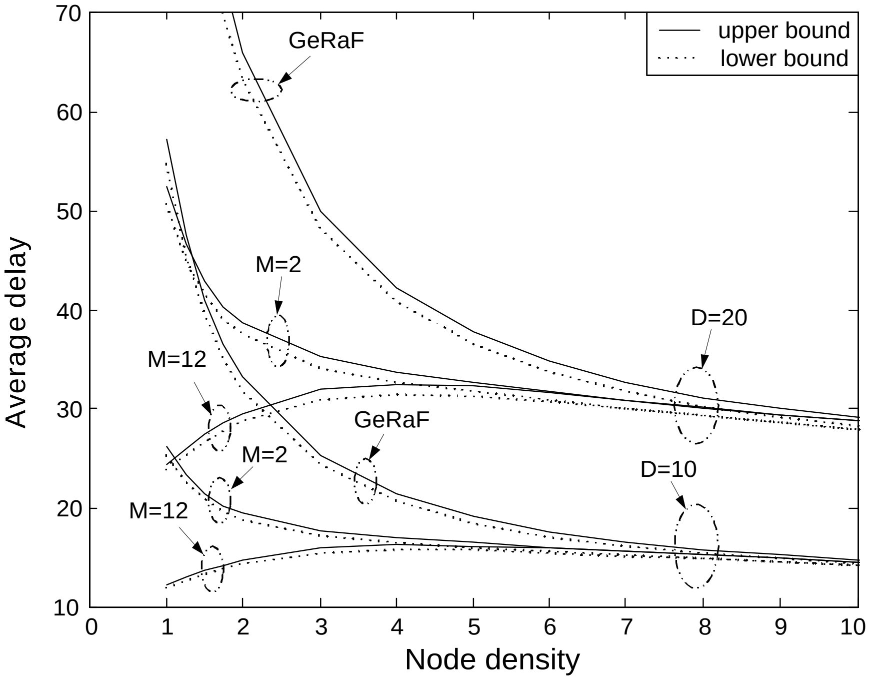

Figure 5.4: The average delay (normalized by τ) of Slow-HARBINGER B under different rate constraints M, $M = 1$ corresponds to GeRaF.

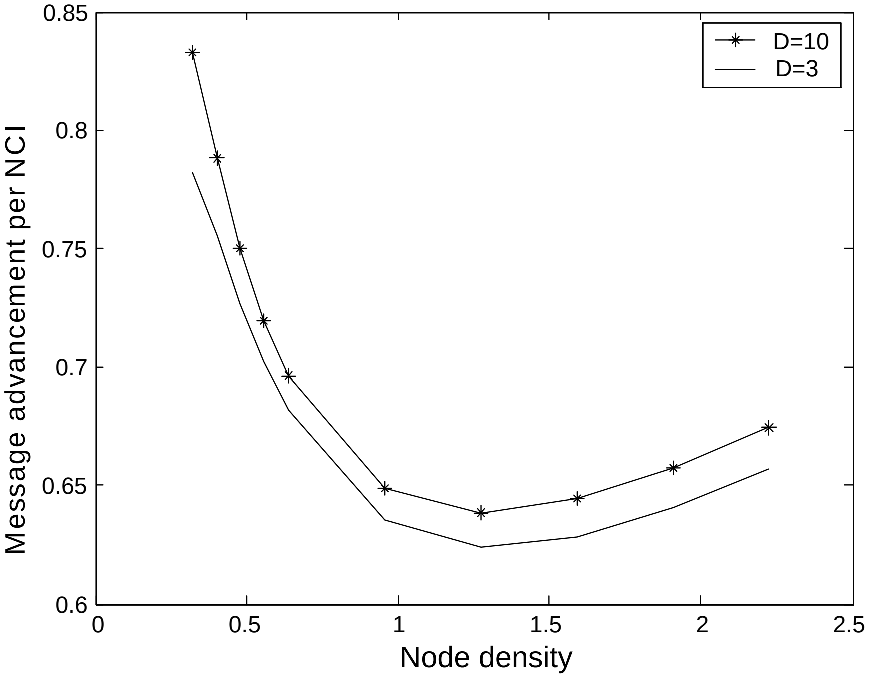

Figure 5.5: The average message progress per NCI for Slow-HARBINGER B under different source/destination separation.

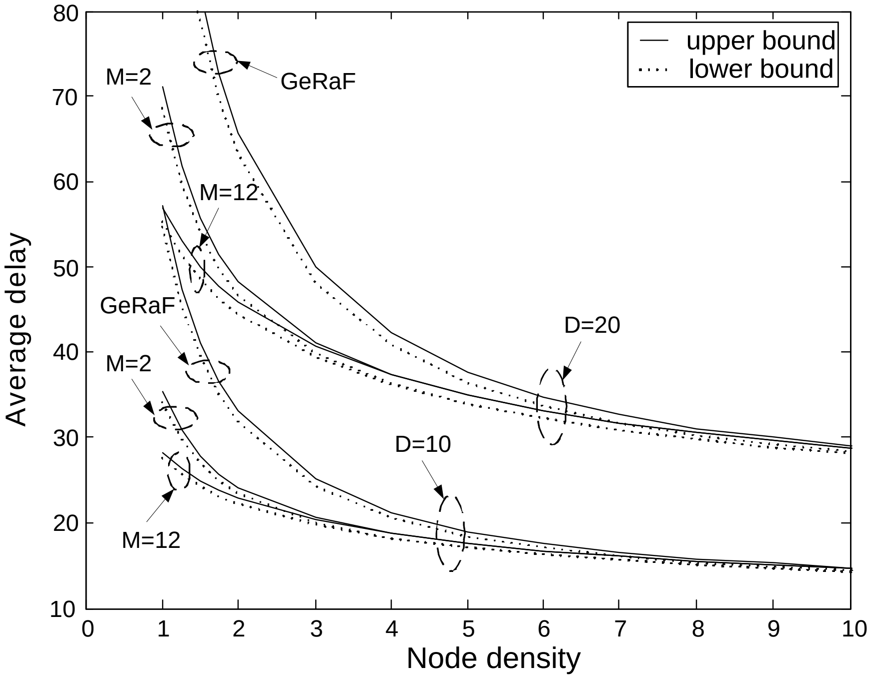

Figure 5.6: The average delay (normalized by τ) of Fast-HARBINGER under different rate constraints M, $M = 1$ corresponds to GeRaF.

to the destination are more likely to be chosen as relay in low density network. Therefore, its corresponding message delay becomes smaller as shown in Fig. 5.4.

Finally, we observe that Fast-HARBINGER has almost the same delay characteristics as that of Slow-HARBINGER B except that with Fast-HARBINGER, the message delay is a monotonically decreasing function with respect to node density even under large rate constraint. The delay performance of HARBINGER indicates that with hybrid-ARQ, nodes are allowed to remain in a sleep state for a relatively longer percentage of time than GeRaF (for the same total node density) while still able to achieve the same delay performance as GeRaF. Alternative, with the same duty cycle of network devices, HARBINGER could significantly reduce the message delay.

In general, it is rather difficult to compare the delay performance of Slow-HARBINGER and Fast-HARBINGER, since their corresponding network coherence times are usually quite different. Here we considered a special case where the data packet length L is much longer than the signalling packet and the network coherence time τ is equivalent to the maximum data packet length allowed per NCI. In particular, in Fast-HARBINGER $\tau = L$, while in Slow-HARBINGER $\tau = ML$. If we further assume that the data packet length L is the same for either Fast-HARBINGER or Slow-HARBINGER, we are able to compare the message delay in different versions of HARBINGER in Fig. 5.7 and Fig. 5.8 under different rate constraints $M = 2, 12$ respectively. Noticing the proximity of upper and lower bounds, we only show the message delay lower bound in both figures. Notice that Fast-HARBINGER has better delay performance than either Slow-HARBINGER A

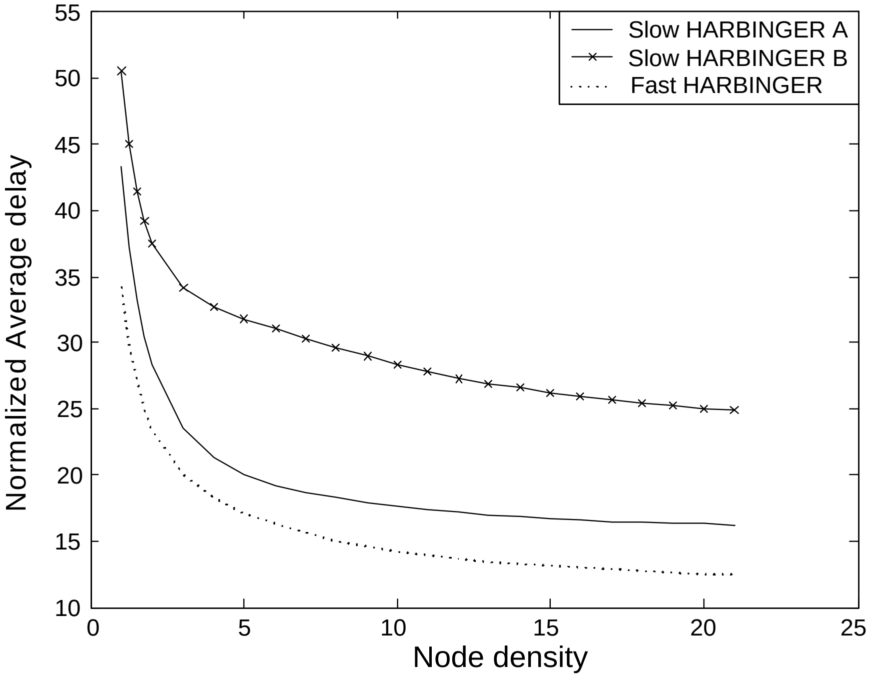

Figure 5.7: The message delay lower bound in different versions of HARBINGER under rate constraint $M = 2$. The average message delay is normalized by data packet length L. The source-destination separation $D = 10$.

or Slow-HARBINGER B, since it not only cuts down the number of NCIs per message but also reduces network coherence time τ by intentionally speeding up the sleep cycle of network devices. Finally, we need to mention that the comparison is based on ideal assumptions, which in some cases is not true. For instance, the message delay should take into account the non-negligible length of signalling packets.

5.6.2 Energy Efficiency

The energy efficiency analysis of HARBINGER is highly dependent on the ratio of energy consumed by signalling packets to the energy consumed by data packets. As mentioned earlier, if we assume that data packets take up a majority of energy dissipation and ideally ignore the energy dissipation of signalling packets, the energy dissipation is linearly proportional to the average number of data packet transmissions per message. Due to the proximity of both bounds, we only plot the lower bound of data packet transmissions per message for Slow-HARBINGER and Fast-HARBINGER in Fig. 5.9, Fig. 5.10, and Fig. 5.11. We observe that in all three figures, GeRaF actually has the best energy efficiency. HARBINGER consumes more energy than GeRaF primarily due to its relatively aggressive packet transmission strategy and non-linear expansion of rate constraint circles. More specifically, GeRaF does not a transmit data packet if there is no relay in the rate constraint circle of $M = 1$ which turns out to be the most energy efficient

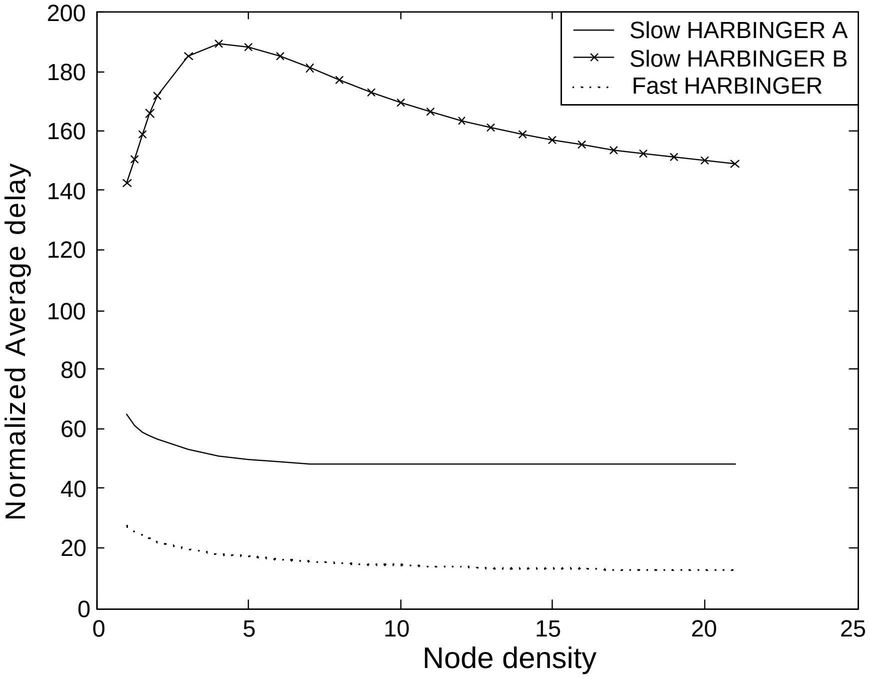

Figure 5.8: The message delay lower bound in different versions of HARBINGER under rate constraint $M = 12$. The average message delay is normalized by data packet length L. The source-destination separation $D = 10$.

circle in HARBINGER. Further notice that unlike the message delay which decreases as the rate constraint increases, the energy consumption of HARBINGER actually increases along with the rate constraint. In particular, with Slow-HARBINGER A the energy dissipation increases significantly as M increases in both dense and sparse networks. Unlike Slow-HARBINGER A, although the energy efficiency of Slow-HARBINGER B and Fast-HARBINGER is worse than GeRaF in low density networks, they all converge to GeRaF in high density networks. In fact, as $\rho \to \infty$, both Slow-HARBINGER B and Fast-HARBINGER asymptotically require $\lfloor D + 1 \rfloor$ data packet transmissions for each message. In addition, as the rate constraint gets fairly large, i.e. $M = 12$, the message delay of Fast-HARBINGER is almost equivalent to the average data packet transmissions per message, indicating that with Fast-HARBINGER there almost always exists at least one relay node within the coverage area at the first NCI of each session.

Fig. 5.12 and Fig. 5.13 further compare the energy efficiency of different versions of HARBINGER under rate constraint $M = 2, 12$. Notice that Slow-HARBINGER A has the worst energy efficiency especially under large rate constraint. In general, Slow-HARBINGER B has almost the same energy efficiency as Fast-HARBINGER with Fast-HARBINGER performing a little better in low density networks.

Altogether we investigated two different network setups where the source and destination are separated by distances of 10 and 20. We observe that both message delay and average data packets

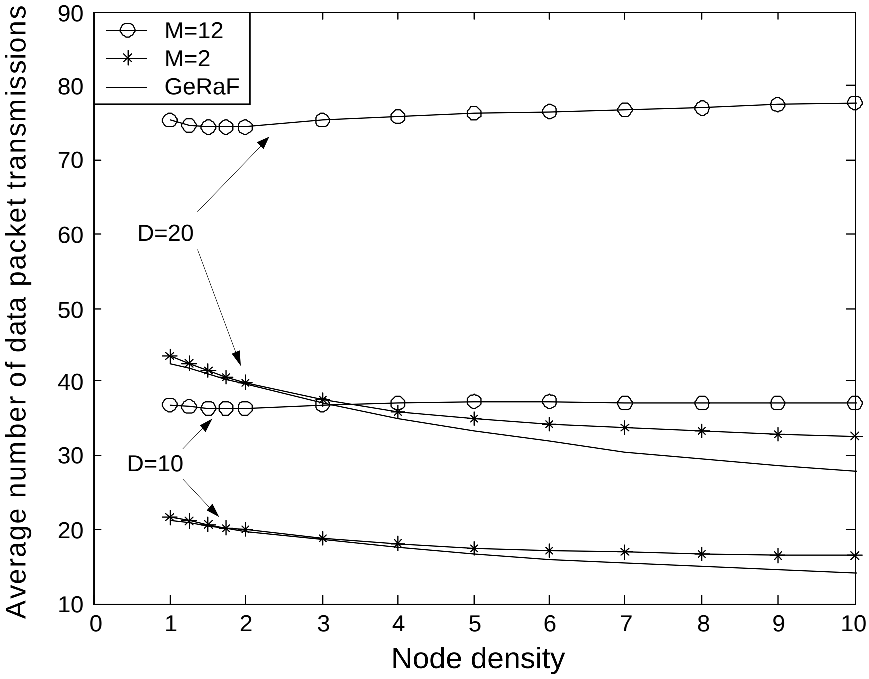

Figure 5.9: The average data packet transmissions per message in Slow-HARBINGER A under different rate constraints M, $M = 1$ corresponds to GeRaF.

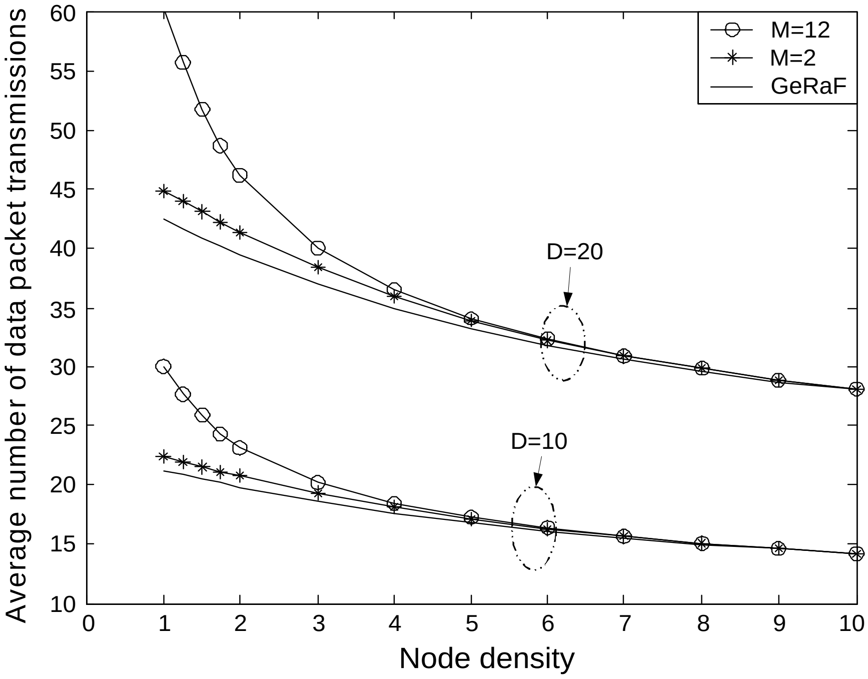

Figure 5.10: The average data packet transmissions per message in Slow-HARBINGER B under different rate constraints M, $M = 1$ corresponds to GeRaF.

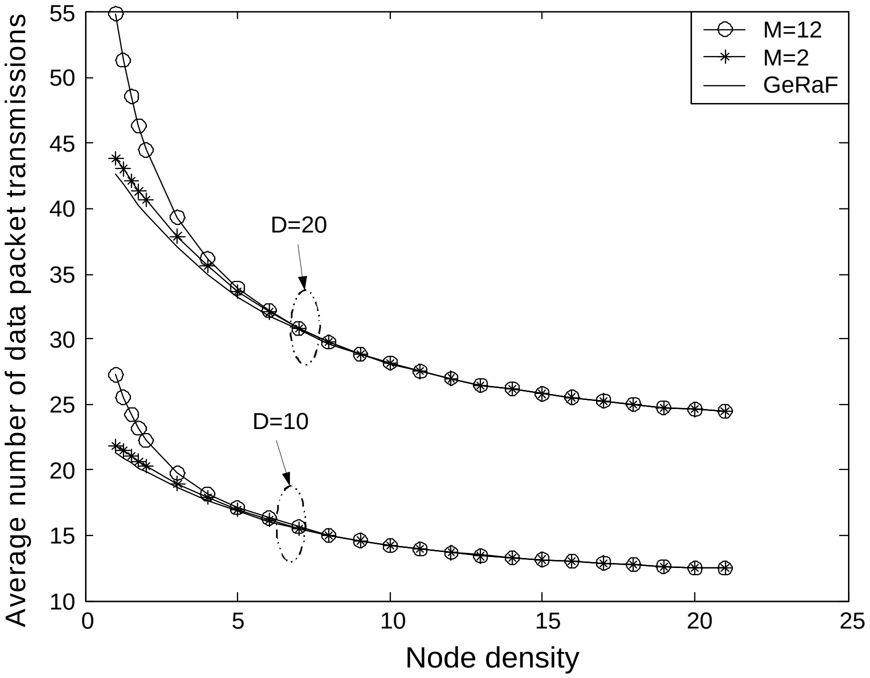

Figure 5.11: The average data packet transmissions per message in Fast-HARBINGER under different rate constraints M, $M = 1$ corresponds to GeRaF.

per message are almost linearly proportional to the separation distance. Therefore, as we increase the transmit SNR, although the normalized radius $\{r_i\}$ will not change, the distance D will be proportionally reduced by a factor of d_1, thus decreasing the delay and data packet transmissions accordingly. Further note that the above bounds are derived under the assumption of memory flushing (after each successful message transmission) which significantly reduces the relaying gain in the protocol. Therefore, in practice, without memory flushing, HARBINGER should perform much better in the sense of both message delay and energy efficiency. Finally, we need to point out that each version of HARBINGER could be most suitable for different sensor network applications. The performance comparison of different HARBINGER schemes is based on idealized assumptions, and thus should be interpreted with caution. In practice, increasing the rate constraint does not necessarily improve the performance of HARBINGER. Rather HARBINGER with a small rate constraint, i.e. $M = 2, 3$, is appropriate for network implementation, since under small rate constraints HARBINGER could dramatically decrease the message delay without a significant increase in the energy dissipation.

5.7 Summary

HARBINGER is an effective cross-layer protocol for ad hoc networks that combines Geographic Random Forwarding with hybrid-ARQ. Different versions of HARBINGER are proposed and their

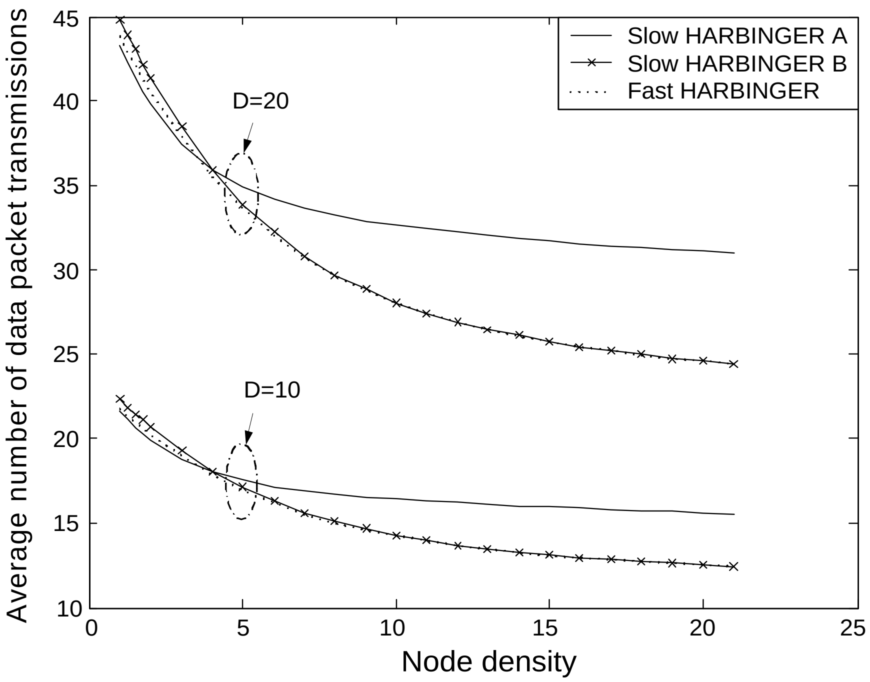

Figure 5.12: The energy efficiency of different versions of HARBINGER with rate constraints $M = 2$ under different source/destination separation.

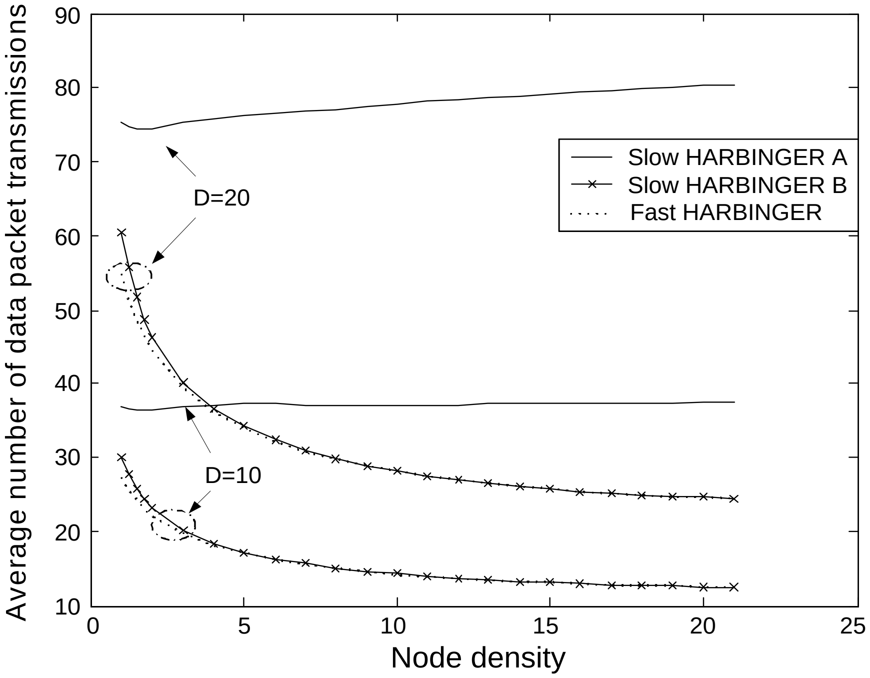

Figure 5.13: The energy efficiency of different versions of HARBINGER with rate constraints $M = 12$ under different source/destination separation.

performance analyzed with respect to message delay and energy efficiency. The analysis presented in this chapter generalizes GeRaF, which corresponds to the specific case that $M = 1$. HARBINGER is especially beneficial over GeRaF in lower density networks when small rate constraint hybrid-ARQ is applied, indicating that a smaller duty-cycle sleep schedule could be used with HARBINGER, thereby increasing the useful lifetime of sensor networks. Alternatively, for the same sleep schedule, HARBINGER allows reduced end-to-end delay compared to GeRaF. In contrast with Slow-HARBINGER, Fast-HARBINGER intentionally changes the network topology prior to each data packet transmission, thereby achieving an additional time diversity benefit in time varying networks. Such benefit is manifested by a significant reduction in the message delay. Notice that the fundamental assumption we made in the numerical analysis is that each network device should flush their memory after each successful message transmission. However, through memory flushing, HARBINGER essentially loses most of its relaying gain/cooperative diversity benefit over conventional multihop routing. Therefore, when implementing HARBINGER in sensor networks, memory flushing should be avoided. Finally, we notice that without memory flushing the model quickly becomes mathematically untractable. Eventually, we will have to rely on Monte Carlo integration to investigate the diversity benefit of HARBINGER in both AWGN and block fading channels.

Chapter 6

Distributed Turbo Coding for Relay Networks

6.1 Introduction

In the previous chapters, we have proposed energy efficient cross-layer protocols for constrained relay networks and studied their information theoretic limit. In this chapter, we propose and analyze a practical coding strategy, namely distributed turbo coding, to approach the performance limits of constrained relay networks. The coding strategies we propose is a particular application of rate compatible punctured turbo codes for relay networks. The major spin that we take in the code design is to minimize the decoder complexity in the relay nodes so that it could be readily applied to low-cost networks.

The remainder of the chapter is organized as follows: section 6.2 gives a brief introduction to turbo codes. Section 6.3 proposes distributed turbo codes for the relay networks and finally section 6.4 draws conclusions.

6.2 Turbo Coding for the Noisy Channel

In his 1948 paper, “A Mathematical Theory of Communication”, Shannon proved that channel capacity is achieved with high probability by a randomly constructed code with arbitrarily long frame size. However, he did not suggest a practical way of constructing such codes. There is a general coding principle initially implied by Shannon, and later further developed by other coding theorists that: *For any discrete input memoryless channel, there exists an n-symbol code of rate R for which the codeword error probability with maximum likelihood decoding is bounded by*

$$P_w(e) \quad = \quad \exp\{-n\Theta(R)\} \tag{6.1}$$

where $\Theta(R)$ is the convex error function. $\Theta(R)$ increases with respect to received power P, and decreases monotonically with respect to R, $0 \leq R \leq C$, C is the channel capacity.

According to (6.1), there are three ways to improve the code performance:

- Decreasing the code rate R will increase the error function $\Theta(R)$. However, one side effect is that it will decrease the spectral efficiency.

- Increasing the signal power P will lift the entire error function $\Theta(R)$, thus $\Theta(R)$ will be increased at any fixed rate R. It is straightforward that this approach will decrease the energy efficiency.

- If $\Theta(R)$ is fixed, so that both energy efficiency and spectral efficiency are preserved, the only way to improve performance is to increase the codeword length n. With maximum likelihood decoding, the decoding complexity of a rate R code is proportional to $\exp\{nR_c\}$, hence the conclusion that the code performance improves at the cost of decoding complexity and latency. In fact, more than fifty years of development in the field of error control coding was characterized by an endless research effort to construct capacity approaching codes of linear complexity increase with respect the code length n.

A major breakthrough came in 1993, when Berrou [11] showed that the combination of parallel concatenated convolutional encoding, interleaving and iterative decoding could come within 0.5 dB of the Shannon capacity for binary modulation. The success of turbo codes is due to their ability to construct a pseudo-random codeword with arbitrarily long framesize, yet still be able to decode with reasonable complexity.

The introduction of turbo codes launched numerous research efforts that permanently changed the way we look at error control coding. Initial efforts to reveal the working mechanism and design principle of turbo codes has discovered a whole family of turbo-like codes based on the same principle of code concatenation and iterative decoding. The name "turbo code" itself, initial used for Berrou's parallel concatenated convolutional codes, has evolved into a general term for a family of concatenated codes, including serial concatenated codes (SCCs) [75], parallel concatenated codes (PCCs) [11], and hybrid concatenated codes (HCCs) [76]. Later research [77] aiming at a more general conceptual understanding of turbo decoding reveals a close connection between graphs, Bayesian networks, and error correction codes, which provide new insight into graph based codes, i.e. Gallager's low density parity check codes (LDPC), and their corresponding decoding algorithms.

What's more important about turbo codes is the application of iterative decoding to approximate maximum likelihood decoding. Most modern communication systems are composed of multiple functional block in cascade. In a conventional receiver, the soft information flows through

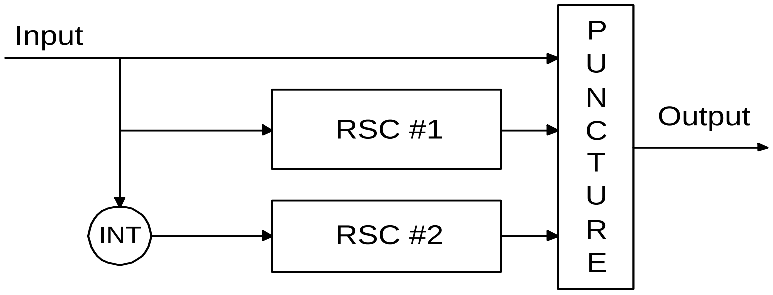

Figure 6.1: A turbo encoder.

each block unilaterally until it reaches the sink. Thus, the corresponding design philosophy for a conventional system is to optimize every single functional block with little regard to how the blocks interact. However, these concatenated blocks could be conceptually considered as a decoder for a generalized serial or parallel concatenated code. According to the iterative decoding principle, the soft information generated from the current block could be used by other blocks to help generate their own soft information. Thus, by iteratively exchanging soft information between multiple blocks, the system performance could be dramatically improved. The aforementioned turbo processing, as a direct extension of turbo decoding, has more or less reshaped the philosophy of modern system design. In particular, joint optimization among multiple blocks, along with turbo processing, will become more prevalent in future communication systems, e.g. turbo equalization[78] and turbo multi-user detection [79] [80], and joint source-channel decoding [81].

6.2.1 Turbo Code Structure

Fig.6.1 shows the encoder of a Parallel Concatenated Convolutional Code (PCCC), made up of two constituent Recursive Systematic Convolutional (RSC) encoders and one pseudo-random interleavers. Both RSC encoders could be either identical (symmetric PCCCs) or different (asymmetric PCCCs [82]). The two constituent encoders will encode two different copies of the same input sequence respectively, one in its natural order, and the other after being interleaved. Both encoders will puncture (a process that deletes selected coded bits at the output of an encoder) the input sequence and only output the parity sequences. The input sequence together with two parity sequences are concatenated to form the final code word. Suppose the constituent code is a rate $\frac{1}{n}$ RSC encoder, then the total code rate for the PCCC is $\frac{1}{2n-1}$. Any desirable code rate could be achieved by properly puncturing the additional parity bits from the final code word.

One way to relate the output of a constituent RSC encoder to its input is through the so called "D transform" [83]. The D transforms for an input sequence $U = [U_0, U_1, U_2, ...U_m]$ is

$U(D) = \sum_{i=0}^{m} U_i D^i, U_i \in [0, 1]$. The output of a rate $\frac{1}{2}$ RSC encoder is a vector

$$(O_1(D), O_2(D)) = U(D) \left(1, \frac{g_2(D)}{g_1(D)}\right) \tag{6.2}$$

where $g_1(D) = \sum_{j=0}^{k_1} g_{1,j} D^j$, and $g_2(D) = \sum_{j=0}^{k_2} g_{2,j} D^j$ are the "D transform" of the encoder's feedback and feedforward generator polynomials, respectively. For notational simplicity, we usually use the coefficients g_i to denote generator polynomials. For instance, $\left(1, \frac{D^4+D^2+1}{D^4+D}\right)$ could be represented by $(1, \frac{10101}{10010})_2$ or $(1, 25/22)_8$.

Notice that with an RSC encoder, $O_1(D)$ is reduced to the input sequence, and thus is termed the systematic output, while $O_2(D)$ is the convolution of the input sequence and the impulse response of an Infinite Impulse Response encoder, and thus is termed the parity output.

6.2.2 Iterative Decoder

Turbo codes use iterative decoding to approximate maximum likelihood decoding, with each constituent decoder employing the maximum a posteriori (MAP) algorithm [84]. The basic structure of an iterative decoder is illustrated in Fig.6.2. Usually the channel observation should be normalized into Log Likelihood Ratio (LLR) form before being fed into each constituent decoder,

$$\Lambda_{Y_i} = \log \frac{P(Y_i|U_i = 1)}{P(Y_i|U_i = 0)} = Re\left\{\frac{2Y_i a_i^*}{\sigma^2}\right\}. \tag{6.3}$$

where Λ_{Y_i} is LLR for channel observation Y_i, Y_i is the i^{th} element of channel observation vector Y, U_i is its corresponding code bit, σ^2 is the noise variance and a_i is the complex channel gain. In AWGN, the channel fading coefficient $a_i = 1$.

The soft information from the output of both decoders could be decomposed into

$$\begin{aligned}
\Lambda_i &= \log \frac{Pr(U_i = 1|Y)}{Pr(U_i = 0|Y)} \\
&= \log \frac{Pr(Y_i|U_i = 1)}{Pr(Y_i|U_i = 0)} + \log \frac{Pr(Y_{k \neq i}|U_i = 1)}{Pr(Y_{k \neq i}|U_i = 0)} + \log \frac{Pr(U_i = 1)}{Pr(U_i = 0)} \\
&= \Lambda_{Y_i} + \Lambda_{e_i} + \Lambda_{U_i}
\end{aligned} \tag{6.2}$$

The third term Λ_{U_i} in (6.2) is the extrinsic input information coming from the other decoder. The second term Λ_{e_i} is the extrinsic output information generated by the current decoder due to the code structure of (6.2). It will later be used as extrinsic input information for the other decoder. Thus, each decoder will take as its input not only channel observation Λ_{Y_i} but also extrinsic input information Λ_{U_i} from the other decoder, and generate its extrinsic output information Λ_{e_i}.

The iteration starts with decoder 1. In particular, decoder 1 is provided with the channel observation of both the systematic sequence and the first parity sequence. Likewise, the second

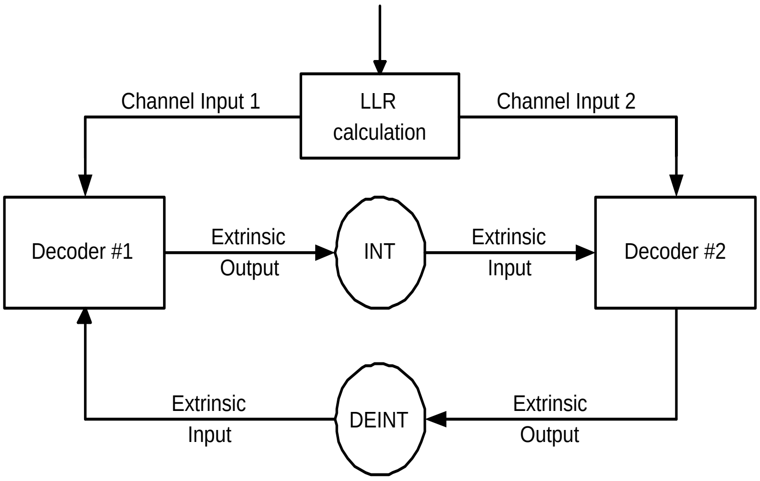

Figure 6.2: A turbo decoder.

decoder is provided with the channel observation of the second parity sequence and the *interleaved* systematic sequence. No extrinsic input information is supplied to decoder 1, since initially the information bit U_i is assumed to be equal likely. After MAP decoding, decoder 1 generates Λ_{e_i}, which is fed into decoder 2 after being interleaved as its extrinsic input information. Similarly, decoder 2 will compute its own Λ_{e_i}, which will be fed into decoder 1 after being deinterleaved as extrinsic input information for the next round of iteration. By iteratively exchanging extrinsic information between two decoders, significant coding gain could be achieved with acceptable system complexity.

6.2.3 Performance Characteristics

The Bit Error Rate (BER) curve of a turbo code could be typically characterized by two distinctive regions. The “waterfall region”, also known as the “turbo cliff”, occurs at low SNR, and has a steep slope if a sufficiently long framesize is used. The “error floor” refers to the BER curve with a rather flattening slope, appearing at moderate to high SNR.

With maximum likelihood decoding, the performance of a linear code is determined by its distance spectrum. On the one hand, the minimum Hamming distance of the code will dominate the BER performance at high SNR, which is formally known as the minimum distance asymptote or free distance asymptote. Therefore the error floor of turbo codes is primarily determined by the codes’ free distance and effective free distance, which is the minimum output weight of a weight 2 input sequence. Turbo codes with a stronger constituent code (in the sense that it uses longer constraint length or primitive polynomials) could increase free distance and effective free distance simultaneously, thus resulting in a better performance in the error floor. However, this characteristic does not necessarily imply a better “turbo cliff”. In particular, bounding techniques indicate that code performance at moderate to low SNR is dominated by the low weight codewords. In conventional code design, engineers are interested in constructing codes with large free distance. However, the design principle of turbo codes follows a totally different strategy. The use of two convolutional encoders in conjunction with the interleaver produces a code that contains very few code words of low weight. In particular, the pseudo-random interleaver reorders the information sequence before it’s encoded by the second parity generator. Thus, even if a low weight parity sequence is generated by the first encoder, due to random interleaving, it is highly unlikely that the second encoder will generate a low weight sequence. Thus, the excellent performance at moderate BER’s is due rather to a drastic reduction in the number of nearest neighbor codewords compared to a convolutional code. This spectral thinning effect, nearly proportional to the framesize N, is discussed in [85][86], and referred to as “interleaver gain”.

At very low SNR, iterative decoding doesn’t work at all, thus the code performance is far away from the maximum likelihood decoding bound. When the channel SNR exceeds a certain threshold, the iterative decoding algorithm starts to converge to the correct codeword, therefore, the BER curve drops dramatically within the turbo cliff area. The convergence of iterative decoding (the position of turbo cliff) could be accurately predicted with the assistance of an extrinsic information transfer (EXIT) chart [87]. At certain channel SNR, there is an EXIT curve revealing the relationship between the entropy of extrinsic input information Λ_{U_i} and extrinsic output information Λ_{e_i} in each constituent decoder. According to the iterative decoding principle, two curves can be drawn on the same plot with the output entropy of each decoder becoming the input entropy of the other. The decoding trajectory of the extrinsic information during iterative decoding follows a staircase path along the EXIT curves of the two decoders. Convergence occurs when the two curves no longer overlap, i.e. when a “decoding tunnel” opens up. Based on EXIT chart technique, some new RSC codes were found to be able to construct turbo codes with “turbo cliff” occurring at lower signal-to-noise ratios.

6.3 Distributed Turbo Coding

6.3.1 Constrained Relay Channel

We now turn our attention to a practical coding method that can approach the information-theoretic performance limits of constrained relay networks. We start with the simplest three-terminal relay networks with $M = 2$. In fact, as $M = 2$, relay networks are reduced to the constrained relay channel. We further assume that the relay channel is operating in a decode-forward mode. The source and relay each employ a very simple code, in this case a two-state

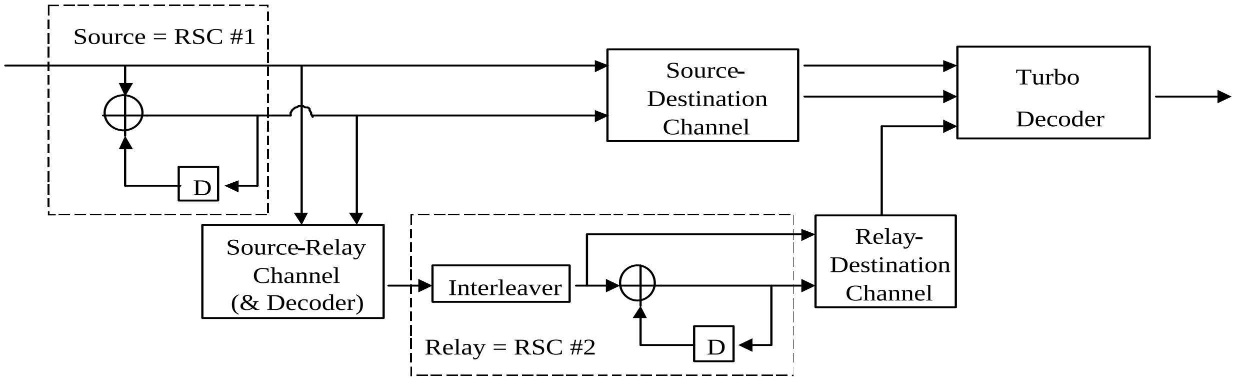
Source = RSC #1
D
Source-Destination Channel
Turbo Decoder
Source-Relay Channel (& Decoder)
Interleaver
Relay-Destination Channel
Relay = RSC #2
D

Figure 6.3: When an interleaver separates source from relay, the relay channel contains a turbo code.

rate 1/2 recursive systematic convolutional (RSC) code with octal generators polynomial $(1, 2/3)_8$, respectively (see Fig. 6.3). After encoding, the signal is BPSK modulated. A conventional decode-forward relay with repetition coding will detect the RSC encoded signal and re-encode it with an identical RSC encoder. The destination will receive two versions of the same code word, one directly from the source and the other from the relay. The two signals may be MRC combined and the message detected with a Viterbi decoder.

The new twist in our proposed scheme is to add an *interleaver* to the relay, as shown in Fig. 6.3. If the relay interleaves its estimate of the source's data prior to RSC encoding, then the source and relay have cooperatively constructed a *distributed* turbo code. Recall that with a turbo code, or parallel concatenated convolutional code (PCCC), the data is recursively encoded twice, first in its natural order and again after being interleaved [11]. Thus, the uninterleaved encoding is present in the source-destination path, while the interleaved encoding is present in the relay-destination path. The destination can detect the code iteratively by using a standard turbo decoder [11]. Although the turbo decoder adds some complexity at the destination, the complexity is still reasonable since the constituent encoders only have two states. While this construction maintains the diversity benefit of relaying, the coding gain is far superior than that of a single RSC observed over two independent channels. This extra coding gain is due to the *interleaving* gain of the turbo code construction and the *turbo processing* gain of the iterative decoder [88]. Conceptually, conventional decode-forward relaying corresponds to MRC-based relaying discussed in chapter 3, while distributed turbo coding corresponds to relaying with incremental redundancy. We have already discussed the information theoretic limits of both schemes in both AWGN and block fading channels in chapter 3. Distributed turbo codes are proposed to approach this limit.

The concept of distributed turbo coding is rather broad. For instance, if the relay re-encodes the detected data with a $(1, 2/3)_8$ RSC encoder prior to interleaving and DPSK encoding, then the relay channel contains a distributed *serial* concatenated convolutional code (SCCC). Also, the extension of distributed turbo coding to the constrained multiple-relay channel and to the cooperative diversity case is straightforward. In particular, when more than two blocks may be transmitted through a multiple-relay channel, the proposed coding strategy becomes a distributed *multiple* turbo code [89].

In essence, the proposed coding approach is a particular instance of a rate compatible punctured turbo (RCPT) code [90]. With RCPT codes, the source information is encoded with a rate R turbo code (termed the mother code). During s_1, the source will puncture the mother code and broadcast a rate R/α punctured turbo code to the relay as well as to the destination. If the relay correctly decodes the first block, it will calculate the parity bits that have been punctured and forward them to the destination during s_2. A distributed turbo code is a RCPT with a particular puncturing pattern such that block one is formed by puncturing out all bits except those created by the upper (uninterleaved) RSC encoder, while the second block contains only the bits created by the lower (interleaved) RSC encoder. The systematic bits could be included in just the first block ($R = 1/3$ and $\alpha = 2/3$) or they could be included in both blocks ($R = 1/4$ and $\alpha = 1/2$). This simple twist in the puncturing pattern is critical in reducing the relay complexity, since it allows the relay to decode the source transmission by a much less complex Viterbi decoder (otherwise an iterative decoder would be needed). However, the disadvantage to this scheme is that the path from source to relay is only protected by a weak RSC code. Thus, this strategy is most suited for situations when the relay is closer to the source than it is to the destination. This constraint on network topology could be partially relieved by using a much stronger RSC code at the source to improve the reliability of the source-destination link, along with a relatively weak RSC code at the relay. This will result in an asymmetric turbo code construction [82].

A simulation campaign was carried out to investigate the performance of distributed turbo coding. In all simulations, data was grouped into frames of length 512 bits. Initially the simple rate 1/2 $(1, 2/3)_8$ RSC code was used to construct a distributed turbo code. Later, more complex RSC codes were used in the code construction to further improve performance.

In the simulations, the transmit SNR of the source (Γ_s) and relay (Γ_r) were varied independently, and it was noted which (Γ_s, Γ_r) pairs achieved a target source-destination frame error rate (FER) of 10^{-2}. We consider two network topologies: (i) the relay is located halfway between the source and destination, and (ii) the relay is 1 m away from the source and 9 m away from the destination. The corresponding simulation results are shown for topologies (i) and (ii) in Fig. 6.4 and Fig. 6.5, respectively. In particular, we compare the performance of several coding strategies against that of the theoretical bound (3.9) for decode-forward relaying with $r = 1/4$ and $\alpha = 1/2$.

Note that results for the same five codes shown in Fig. 6.4 were previously given for topology (i) in [88], but here we have added the theoretical bound to provide a frame of reference. The curves labelled *distributed rate 1/4 PCCC* and *simple PCCC code* correspond to the code shown in Fig. 6.3. For the curve labelled *RSC* relay, there is no interleaver at the relay. For the curve labelled *distributed rate 1/3 PCCC* the relay only sends its parity output. For the curves labelled *distributed rate 1/4 SCCC* and *simple SCCC code* the relay re-encodes the detected message bits with the rate 1/2 RSC prior to interleaving. In addition, results for two stronger codes are shown in Fig. 6.5: a rate 1/4 PCCC with generator polynomials $(1, 15/13)_8$ labelled *stronger PCCC code*, and a rate 1/4 SCCC with an inner code polynomial $(1, 2/3)_8$ and an outer code polynomial $(1, 35/23)_8$ labelled *stronger SCCC code*. Fig. 6.5 indicates that although constructed with the simplest RSC code, simple distributed SCCC is only 4.5 dB away from the theoretic limit. With stronger codes, distributed turbo coding (*stronger SCCC code*) could achieve an extra 2 dB gain in energy efficiency, and therefore it is about 2.5 dB away from the theoretical limit.

In Fig. 6.6, we further compare the performance of a distributed PCCC and its corresponding RCPT counterpart (labelled *stronger RCPT code*). Both codes have the same generator polynomial $(1, 15/13)_8$ and code rate $R = 1/4$, but the RCPT sends half of each of the upper and lower encoders' parity bits in each block. The RCPT code outperforms the distributed PCCC by about 1 dB. With the RCPT code, the relay has to use an iterative decoder to detect the source transmission, while distributed turbo coding only requires a simple Viterbi decoder in the relay. Therefore, the extra 1 dB coding gain comes at the price of relay complexity. In addition, we show the performance of the rate 1/4 RCPC code with generator $(23, 35, 27, 33)_8$ used for cooperative coding in [66]. Its corresponding SNR contour indicates that the RCPC code is 5.25 dB away from the theoretic limit, and thus the stronger PCCC code outperforms it by about 2.5 dB.

6.3.2 Relay Networks with larger Rate Constraint

When the concept of distributed turbo coding is extended to the relay networks with larger rate constraints, the result is a distributed *multiple* turbo code [89]. In our simulations, we assume that the broadcast channel from source to the multiple relays are always reliable (which is likely when the relays are clustered close to the source), and that through perfect power control, the destination's average received SNR from the source and multiple relays are identical.

A total of seven scenarios were simulated, one for a direct RSC encoded transmission (no relay), and then a pair of simulations for each of $K_r = 1, 2, 4$ relays. The corresponding relay networks has rate constraint $M = K_r + 1$. When multiple relays are present, two strategies were simulated, an RSC code with diversity combining (each relay uses the same RSC encoder and no interleaver; MRC combining and Viterbi decoding is performed at the destination) and a parallel multiple turbo

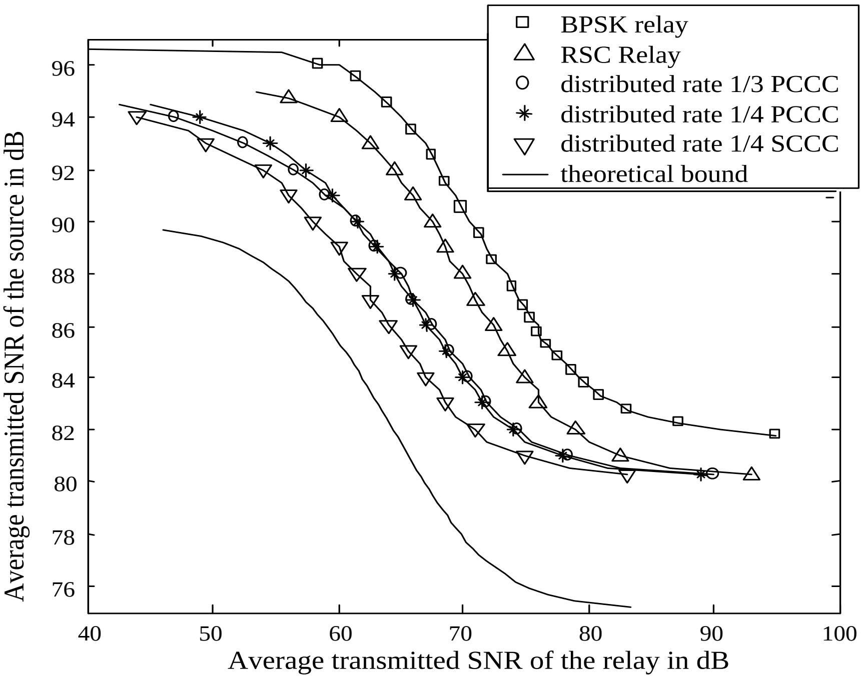

Figure 6.4: Minimum transmit SNR at source and relay required to achieve an end-to-end FER of 10^{-2} when the relay is halfway between the source and destination.

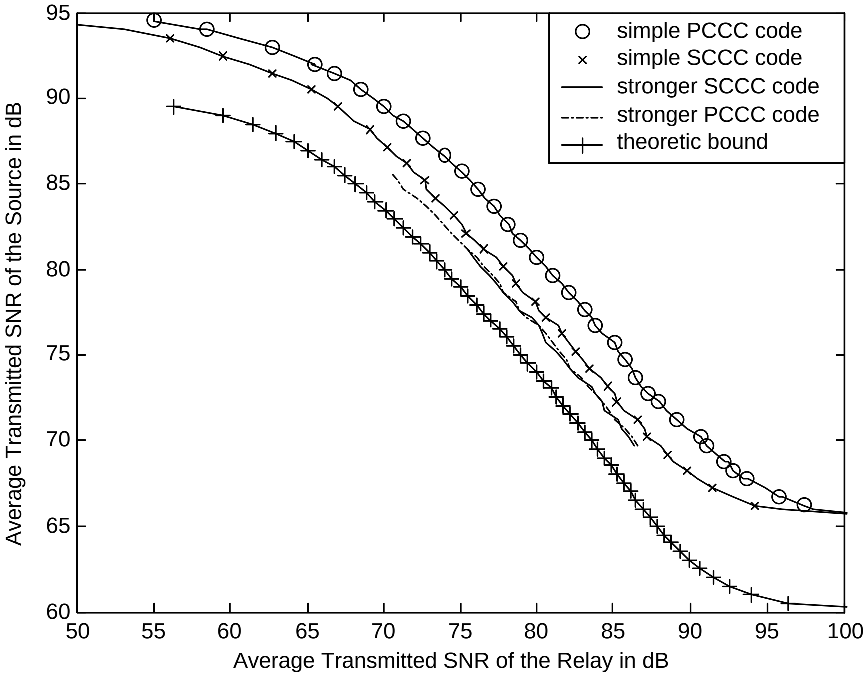

Figure 6.5: Minimum transmit SNR at source and relay required to achieve an end-to-end FER of 10^{-2} when the relay is 1 m away from the source and 9 m away from the destination.

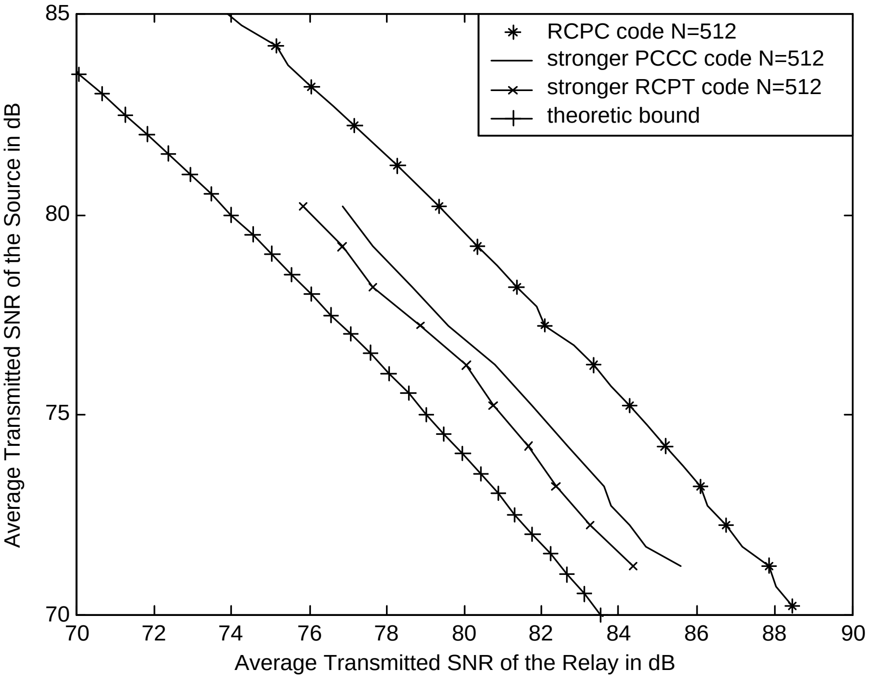

Figure 6.6: Minimum transmit SNR contours of different coding techniques to achieve an end-to-end FER of 10^{-2} when the relay is 1 m away from the source and 9 m away from the destination.

code (each relay interleaves the decoded data with a unique interleaving pattern before differential encoding; a turbo decoder with $K_r + 1$ soft-in/soft-out modules is used at the destination). In each case, the RSC encoders use generator $(1, 2/3)_8$ and the relays only transmit the parity output. Notice that the time slots are nonidentical, since the source broadcasts a rate 1/2 RSC and each relay transmits a rate 1 code. The overall code rate is $R = \frac{1}{K_r+2}$. By varying K_r, we can use the direct transmission results to benchmark the pure diversity gain due to the multiple relay paths, so that the additional interleaving gain of using a multiple turbo code can be easily isolated.

With multiple RSC relaying, a total of 4.7 dB extra diversity gain is be achieved when the number of relays K_r increases from 1 to 4. With distributed multiple turbo codes, the additional interleaving gain increases from 3 dB for single relay ($K_r = 1$) to nearly 6 dB for multiple relays ($K_r = 4$). Compared with the RSC encoded direct link transmission, distributed multiple turbo coding ($K_r = 4$) could achieve a total of 18 dB 'cooperative coding' gain. Both diversity gain and interleaving gain tend to yield diminishing marginal benefit with each additional relay.

6.4 Summary

In this chapter, we developed a simple, but efficient, coding technique for the quasi-static relay channel. Compared to the information theoretic limits of the constrained relay channel

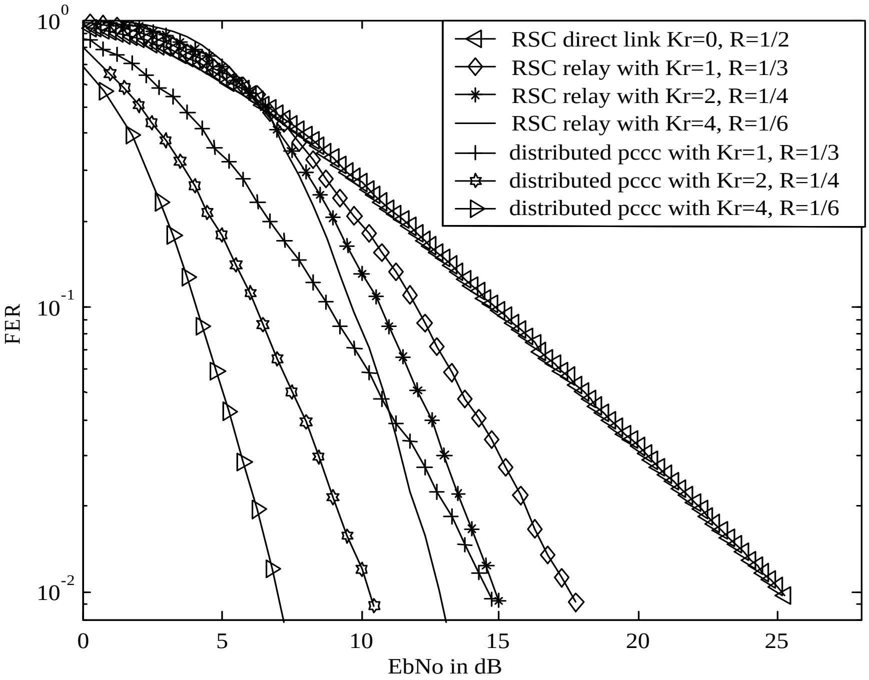

Figure 6.7: FER for distributed multiple turbo codes for relay networks under larger rate constraint under the assumption of perfect source-relay links.

derived in chapter 3, our coding strategies could come within 2.5 dB of the performance limit. A performance comparison between distributed turbo codes and the RCPC code indicates that, with extra interleaving gain, distributed turbo coding is 3 dB more efficient than RCPC code at an end-to-end $FER = 10^{-2}$. Moreover, the extension of distributed turbo coding to relay networks with larger rate constraints results in distributed multiple turbo codes. The performance of such code extension has been investigated and a significant coding gain over the diversity combining based relaying demonstrated.

Chapter 7

Conclusions

7.1 Summary and Conclusions

The major contribution of this dissertation is to characterize the fundamental performance bounds and devise an integrated approach to design, analyze, and implement energy efficient cross-layer protocols for wireless embedded networks. The focus of the study is on a general class of wireless embedded networks that are decomposed into clusters comprised of low cost radio devices including a source, a destination, and one or more relays. Each cluster works cooperatively to convey information from source to destination. The message propagation mechanism of each cluster is modelled as a rate constrained random access relay network where signaling is over a random phase block interference channel, and transmissions from various nodes are non-coherent. Our system model and analysis generalize the research on Gaussian collision channel [21] and the orthogonal relaying work of Laneman et al [45] to more sophisticated multiple relay networks managed by an automatic repeat request (ARQ) protocol. In particular, for relay networks with small rate constraints, we derive closed form bounds on their performance limits in terms of channel capacity and outage probability and further propose new adaptive relaying protocols to improve the network performance. Numerical analysis indicates that even when the rate constraint is as small as $M = 2$, significant energy savings are possible by implementing *distributed* spatial diversity via relaying over block fading channels. We notice that wireless relaying with code combining is about $1 \sim 2$ dB more energy efficient than its diversity combining counterpart, although both schemes have the same diversity order in block fading channels. The achievable relaying gain could be further increased by implementing transmitter diversity with an optimal relay selection strategy in multiple relay channels. However the design and implementation of such strategy require highly coordinated relaying protocols among multiple network devices, which quickly become unwieldy in a large scale network under small rate constraints.

Noticing that the achievable diversity gain in constrained relay networks is primarily restricted by small rate constraints, we increased the rate constraints and ARQ based protocols to fully exploit the spatial diversity with reasonable complexity. Through layer coupling and device cooperation, the resulting cross-layer relaying protocols achieve a better energy-throughput tradeoff than either multihop or direct transmission in relay networks with fixed topology. The diversity benefit of applying relaying to embedded networks is twofold, namely macrodiversity due to nonlinear path-loss effect and microdiversity due to transmitter/receiver diversity in multiple network devices. In relay networks with arbitrary node distribution, the dominance of macrodiversity in throughput and energy efficiency is manifested by the superior performance of 'average relaying' over other relaying protocols.

A key advantage of relaying is that it does not require a network-layer protocol to explicitly select a route through the network a priori. Rather, relaying will adaptively find the best 'path' and will tend to bypass relays that are continually in an outage. Also, power/range control becomes less important in a relaying network. For instance, if the power is set too high in a relay network intermediate relays will simply be 'leapfrogged' and therefore won't need to be used.

A side effect of relaying is that many devices must now listen to each broadcast, in contrast with multihop where only a single device receives each transmission. We investigate the impact of a non-negligible energy cost to receive a transmission and find out that the benefits of relaying begin to diminish when the cost to receive a symbol is on the order of the cost to transmit it. Thus, relaying might be more suitable for transmissions over longer ranges, where transmit power dominates receiver circuit dissipation. Alternatively, for wireless networking over short ranges, the side effect of relaying could be relieved by cautiously defining coverage area and optimal cluster size to avoid excessive receiver energy dissipation.

By further incorporating media access control and routing schemes, the proposed cross-layer protocol could be readily applied to random networks with sleeping cycles. More specifically, the proposed protocol utilize geographic information to jointly perform physical-layer cooperative diversity, data-link-layer hybrid-ARQ retransmission, and network layer relaying/routing, and thus is given a descriptive name Hybrid ARq-Based Intra-cluster GEographically-informed Relaying (HARBINGER). Several versions of HARBINGER with considerably different behaviors have been proposed to meet different requirements of network applications. A unified framework of analysis based on device memory flushing is established to investigate the performance of HARBINGER in AWGN channel. As a generalization of Geographic Random Forwarding (GeRaF), HARBINGER effectively expands its coverage area through type II hybrid-ARQ retransmission, thus results in a dramatic reduction in message delay especially in low density networks. Accordingly, a smaller duty-cycle sleep schedule could be used for network devices with HARBINGER to increase the useful lifetime of embedded networks. However, HARBINGER is less energy-efficient than GeRaF due to its aggressive data packet retransmission, nonlinear coverage expansion, and above all, memory flushing mechanism. Notice that although memory flushing increases the model tractability, it loses cooperative diversity/relaying benefit in densely deployed networks, thus should be avoided in network implementation. Without memory flushing, HARBINGER harvests cooperative diversity and becomes more advantageous for embedded networks under energy constraint and block fading.

Finally, we develop simple coding strategies inspired by the turbo principle to approach the information theoretic limits of the constrained relay networks. The major spin that we take in the code design is to minimize the decoder complexity in the relay nodes so that it could be readily applied to low-cost networks. For embedded networks under small rate constraints, we proposed distributed turbo codes to approach the performance limit within 2.5 dB. The extension of distributed turbo coding to relay networks with larger rate constraints results in distributed multiple turbo codes. With distributed multiple turbo codes, significant coding gain over conventional (diversity combining based) relaying is achievable in embedded networks in block fading channels.

7.2 Future Work

We believe the research work documented in this dissertation could be further extended at least in the following two aspects:

To increase tractability, the HARBINGER analysis is based on an AWGN channel model and memory flushing mechanism. However it does not reflect the cooperative diversity benefit of HARBINGER in embedded networks. More importantly, AWGN is not an accurate signal propagation model of embedded networks. As mentioned in chapter 2, the message propagation of each network cluster is over a random phase block interference channel. Current mathematical framework quickly becomes untractable under these practical constraints. To investigate the cooperative diversity/relay benefit of HARBINGER in block interference channel, a straightforward approach is to apply Monte Carlo integration. However, Monte Carlo integration/simulation is heavily dependent on network configurations and is computationally intensive (time consuming). Therefore a unified framework of analysis that incorporates both device memory should be established to provide maximum tractability for relay networks over both AWGN and block fading.

We have proposed an incremental redundancy based coding strategy, namely distributed turbo coding, to approach the information-theoretic limits of constrained relay networks. It is straightforward that other capacity approaching codes based on incremental redundancy should be developed and incorporated into ARQ based relaying protocols, including rate compatible punctured Low Density Parity Check (LDPC) codes and rate compatible punctured Irregular Repeat Accumulative (IRA) codes. Notice that in constrained relay networks, messages are relayed over block fading channel, therefore the focus of code design should be on a special class of rate compatible codes whose performance is optimized over block fading channels [33].

References

[1] D. Estrin, R. Govindan, and J.Heidemann, “Embedding the Internet,” *Communications of the ACM*, vol. 43, pp. 39–41, May 2000.

[2] National Research Council (Computer Science and Telecommunications Board), *Embedded, Everywhere.* Washington, DC: National Academy Press, 2001.

[3] J. A. Gutierrez, M. Naeve, E. Callaway, M. Bourgeois, C. Mitter, and B. Heile, “IEEE 802.15.4: A developing standard for low-power low-cost wireless personal area networks,” *IEEE Network*, vol. 15, pp. 12–19, Sept.-Oct. 2001.

[4] Institute for Electrical and Electronics Engineers, “Standard for part 15.4: Wireless medium access control (MAC) and physical layer (PHY) specifications for low rate wireless personal area networks (lr-wpans),” *P 802.15.4/D18*, Feb. 2003. draft.

[5] A. J. Goldsmith and S. B. Wicker, “Design challenges for energy-constrained ah hoc wireless networks,” *IEEE Wireless Communications*, pp. 8–27, Aug. 2002.

[6] A. Ephremides, “Energy concerns in wireless networks,” *IEEE Wireless Commun.*, vol. 9, pp. 46–69, Aug. 2002.

[7] M. Zorzi and R. R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Multihop performance,” *IEEE Trans. Mobile Comp.*, vol. 2, pp. 337–348, Oct. 2003.

[8] C. E. Shannon, “A mathematical theory of communication,” *Bell Sys. Tech. J.*, vol. 27, pp. 379–423 and 623–656, 1948.

[9] G. Ungerboeck, “Channel coding with multilevel/phase signals,” *IEEE Trans. Inform. Theory*, vol. 28, pp. 55–67, Jan. 1982.

[10] S. Benedetto, D. Divsalar, D. Montorsi, and F. Pollara, “Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding,” *IEEE Trans. Inform. Theory*, vol. 44, pp. 909–926, May 1998.

[11] C. Berrou, A. Glavieux, and P. Thitimasjshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes(1),” in *Proc., IEEE Int. Conf. on Commun.*, (Geneva, Switzerland), pp. 1064–1070, May 1993.

[12] R. G. Gallager, *Low-Density Parity-Check Codes.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1960.

[13] G. J. Foschini, “Layered space-time architecture for wireless communication in a fading enviroment when using multi-element antennas,” *Bell Labs Technical Journal*, vol. 1, no. 2, pp. 41–59, 1996.

[14] L. Zheng and D. Tse, “Diversity and multiplexing: A fundamental tradeoff in multiple antenna channels,” *IEEE Trans. Inform. Theory*, vol. 49, pp. 1073–1096, May 2003.

[15] T. M. Cover and A. A. El Gamal, “Capacity theorems for the relay channel,” *IEEE Trans. Inform. Theory*, vol. 25, pp. 572–584, Sept. 1979.

[16] S. Chung and A. J. Goldsmith, “Degree of freedom in adaptive modulation: A unified view,” *IEEE Trans. Commun.*, pp. 1561–1571, Sept. 2001.

[17] A. Goldsmith and P. Varaiya, “Capacity of fading channels with cahnnel side information,” *IEEE Trans. Inform. Theory*, pp. 1986–1992, Nov. 1997.

[18] N. Bambos, “Toward power-sensitive network architectures in wireless communications: concepts, issues, and design aspects,” *IEEE Personal Commun.*, pp. 50–59, June 1998.

[19] C. Chiasserini and R. Rao, “Energy-efficient battery management,” *IEEE J. Select. Areas Commun.*, pp. 1235–1245, July 2001.

[20] N. Abramson, “The ALOHA system — another alternative for computer communications,” in *Proc. AFIPS Conf.*, (Las Vegas, NV), pp. 281–285, 1970.

[21] G. Caire and D. Tuninetti, “The throughput of hybrid-ARQ protocols for the Gaussian collision channel,” *IEEE Trans. Info. Theory*, vol. vol. IT-47, pp. pp. 1971 – 1988, July, 2001.

[22] A. MacKenzie and S. Wicker, “Selfish users in ALOHA: A game-theoretic approach,” in *Proc. IEEE Veh. Tech. Conf. (VTC)*, pp. 1354–1357, Oct. 2001.

[23] D. Bertsekas and R. Gallager, *Data Networks*. 1992.

[24] K. Parhi and R. Ramaswami, “Distributed scheduling of broadcasts in a radio network,” in *IEEE INFOCOM*, pp. 497–504, Mar. 1989.

[25] D. Goodman, “Packet reservation multiple access for local wireless communications,” *IEEE Trans. Commun.*, Aug. 1989.

[26] E. M. Royer and C. K. Toh, “A review of current routing protocols for ad hoc mobile wireless networks,” *IEEE Personal Commun. Mag.*, vol. 6, pp. 46–55, Apr. 1999.

[27] S. Toumpis and A. J. Goldsmith, “Capacity regions for wireless ad hoc networks,” *IEEE Trans. Wireless Commun.*, vol. 2, pp. 736–748, July 2003.

[28] R. Gallager, “Energy limited channels: coding, multiple-access, spread spectrum,” in *Conf. Info. Sys. Sci.*, Mar. 1988.

[29] S. Verdú, “On channel capacity per unit cost,” *IEEE Trans. Inform. Theory*, pp. 1019–1030, Sept. 1990.

[30] H. Mandyam and A. J. Goldsmith, “Capacity of finite energy channels,” in *Allerton Conf. Commun. Cntl. Comp.*, Oct. 2001.

[31] W. Stark, H. Wang, A. Worthen, S. Lafortune, and D. Teneketzis, "Low-energy wireless communication network design," *IEEE Wireless Communications*, pp. 60–72, August 2002.

[32] E. C. van der Meulen, "Three-terminal communication channels," *Adv. Appl. Prob*, vol. 3, pp. 120,154, 1971.

[33] A. Stefanov and E. Erkip, "Cooperative coding for wireless networks," in *Proc. IEEE Conf. on Mobile and Wireless Commun. Networks*, (Stockholm, Sweeden), Sept. 2002.

[34] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Resource management in energy-limited, bandwidth-limited, transciever-limited wireless networks for session-based multicasting," *Computer Networks*, vol. 39, pp. 113–131, 2002.

[35] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, pp. 388–404, Mar. 2000.

[36] P. Gupta and P. R. Kumar, "Towards an information theory of large networks: An achievable rate region," *IEEE Trans. Inform. Theory*, vol. 49, pp. 1877–1894, Aug. 2003.

[37] M. Grossglauser and D. N. C. Tse, "Mobility increases the capacity of ad-hoc wireless networks," *IEEE/ACM Trans. on Networking*, vol. 10, pp. 477–486, Aug. 2002.

[38] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inform. Theory*, 2002. to appear.

[39] B. Schein and R. Gallager, "The Gaussian parallel relay network," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, (Sorrento, Italy), p. 22, June 2000.

[40] M. Gastpar and M. Vetterli, "On the capacity of wireless networks: The relay case," in *Proc. INFOCOM*, pp. 1577–1586, 2002.

[41] A. Høst-Madsen, "On the capacity of wireless relaying," in *Proc. IEEE Veh. Tech. Conf. (VTC)*, (Vancouver, BC), Sept. 2002.

[42] G. Kramer, M. Gastpar, and P. Gupta, "Capacity theorems for wireless relay channels," in *Proc. Allerton Conf. Commun., Control, Computing*, Nov. 2003.

[43] H. El Gamal, "On the scaling laws of dense wireless sensor networks," in *Proc. Allerton Conf. Commun., Control, Computing*, (Allerton, IL), Oct. 2003.

[44] M. Khojastepour, A. Sabharwal, and B. Aazhang, "On the capacity of 'cheap' relay networks," in *Conf. on Information Sciences and Systems*, (Baltimore, MD), Apr. 2003.

[45] J. N. Laneman and G. W. Wornell, "Exploiting distributed spatial diversity in wireless networks," in *Proc. Allerton Conf. Commun., Control, Computing*, (Allerton, IL), Oct. 2000.

[46] J. N. Laneman and G. W. Wornell, "Distributed space-time coded protocols for exploiting cooperative diversity in wireless networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, (Taipei, Taiwan), Nov. 2002.

[47] S. Sendonaris, E. Erkip, and B. Aazhang, "Increasing uplink capacity via user cooperation diversity," in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, (Cambridge, MA), p. 156, Aug. 1998.

[48] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity — part I: System description," *IEEE Trans. Commun.*, vol. 51, pp. 1927–1938, Nov. 2003.

[49] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity — part II: Implementation aspects and performance analysis," *IEEE Trans. Commun.*, vol. 51, pp. 1939–1948, Nov. 2003.

[50] L. Ozarow, S. Shamai, and A. D. Wyner, "Information theoretic considerations for cellular mobile radio," *IEEE Trans. Veh. Tech.*, vol. 43, pp. 359–378, May 1994.

[51] R. Knopp and P. A. Humblet, "On coding for block fading channels," *IEEE Trans. Inform. Theory*, vol. 46, pp. 189–205, Jan. 2000.

[52] E. Biglieri, J. Proakis, and S. Shamai, "Fading channels: Information-theoretic and communications aspects," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2619–2692, Oct. 1998.

[53] R. J. McEliece and W. E. Stark, "Channels with block interference," *IEEE Trans. Inform. Theory*, vol. 30, pp. 44–53, Jan. 1984.

[54] G.Caire, E. Leonardi, and E. Viterbo, "Modulation and coding for the Gaussian collision channel," *IEEE Trans. Inform. Theory*, vol. 46, pp. 2007–2026, Sept. 2000.

[55] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Commun.*, vol. 6, pp. 311–335, Mar. 1998.

[56] R.Wolff, *Stochastic Modeling and the Theory of Queues.* Prentice-Hall, 1999.

[57] T. E. Hunter and A. Nosratinia, "Performance analysis of coded cooperation diversity," in *Proc. IEEE Int. Conf. on Commun. (ICC)*, (Anchorage, AK), May 2003.

[58] S. Wicker, *Error Control Systems for Digital Communications and Storage.* Englewood Cliffs, NJ: Prentice Hall, Inc., 1995.

[59] T. S. Rappaport, *Wireless Communications: Principles and Practice.* Upper Saddle River, NJ: Prentice Hall PTR, second ed., 2002.

[60] G. Lauer, "Packet-radio routing," in *Routing in Communications Networks* (M. Steenstrup, ed.), ch. 11, pp. 351–396, Englewood Cliffs, NJ: Prentice Hall, 1995.

[61] M. C. Valenti and B. Zhao, "Distributed turbo codes: Towards the capacity of the relay channel," in *Proc. IEEE Veh. Tech. Conf. (VTC)*, (Orlando, FL), Oct. 2003.

[62] J. N. Laneman and G. W. Wornell, "Energy-efficient antenna sharing and relaying for wireless networks," in *IEEE Wireless Commun. and Networking Conf.*, (Chicago), pp. 7–12, Sept. 2000.

[63] B. Zhao and M. C. Valenti, "Some new adaptive protocols for the wireless relay channel," in *Proc. Allerton Conf. Commun., Control, Computing*, (Monticello, IL), Oct. 2003.

[64] M. Gastpar and M. Vetterli, "On the capacity of wireless networks: The relay case.," in *IEEE Inforcom*, (New York), June 2002.

[65] J. N. Laneman, *Cooperative diversity in wireless networks: Algorithms and architectures.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, Aug. 2002.

[66] T. Hunter and A. Nosratinia, "Performance analysis of coded cooperation diversity," *Proc. IEEE Int. Conf. on Commun. (ICC)*, 2003.

[67] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* Wiley, 1991.

[68] J. N. Laneman and G. W. Wornell, "Distributed space-time coded protocols for exploiting cooperative diversity in wireless networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, (Taipei, Taiwan), Nov. 2002.

[69] M. O. Hasna and M. S. Alouini, "End-to-end performance analysis of two-hop relayed transmissions over Rayleigh fading channels," in *Proceedings of IEEE Vehicular Technology Conference (VTC Fall'2002)*, (Vancouver, British Columbia, Canada), 2002.

[70] J. Hagenauer, "Rate compatable punctured convolutional codes (RCPC-codes) and their application," *IEEE Trans. Commun.*, vol. 36, pp. 389–400, 1988.

[71] M. C. Valenti and N. Correal, "Exploiting macrodiversity in dense multihop networks and relay channels," in *IEEE Wireless Commun. and Networking Conf.*, (New Orleans, LA), Mar. 2003.

[72] A. Chockalingam and M. Zorzi, "Energy efficiency of media access protocols for mobile data networks," *IEEE Trans. Commun.*, vol. 46, pp. 1418–1421, Nov. 1998.

[73] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *IEEE INFOCOM*, (New York, NY), June 2002.

[74] R. Min, M. Bhardwaj, and S. H. C. etc, "Energy-centric enabling technologies for wireless sensor networks," *IEEE Wireless Communications*, Aug. 2002.

[75] S. Benedetto, G. Montorsi, D. Divsalar, and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding," *JPL TDA Progress Report*, vol. 42, Aug. 1996.

[76] D. Divsalar and F. Pollara, "Serial and hybrid concatenation codes with applications," in *Proc., Int. Symp. on Turbo Codes and Related Topics*, (Brest, France), pp. 80–87, Sept. 1997.

[77] R. J. McEliece, D. J. C. MacKay, and J.-F. Cheng, "Turbo decoding as an instance of Pearl's 'belief propagation' algorithm," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 140–152, Feb. 1998.

[78] D. Raphaeli and Y. Zarai, "Combined turbo equalization and turbo decoding," *IEEE Commun. Letters*, vol. 2, pp. 107–109, April 1998.

[79] H. Poor, "Turbo multiuser detection: An overview," in *Proc. IEEE Int. Symp. on Spread Spectrum Techniques and Applications (ISSSTA)*, (Newark, NJ), pp. 583–587, Sept. 2000.

[80] D. Reynolds and X. Wang, "Turbo multiuser detection with unknown interferers," *IEEE Trans. Commun.*, pp. 616–622, Apr. 2002.

[81] J. Hagenauer, "Source-controlled channel decoding," *IEEE Trans. Commun.*, vol. 43, pp. 2449–2457, Sep. 1995.

[82] O. Y. Takeshita, O. M. Collins, P. C. Massey, and D. J. Costello, "Asymmetric turbo-codes," in *Proc., IEEE Int. Symp. on Inform. Theory*, (Cambridge, MA), p. 179, Aug. 1998.

[83] J. G. D. Forney, “Convolutional codes I: Algebraic structure,” *IEEE Trans. Inform. Theory*, vol. 16, pp. 720–738, Nov. 1970.

[84] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inform. Theory*, vol. 20, pp. 284–287, Mar. 1974.

[85] S. Benedetto and G. Montorsi, “Unveiling turbo codes: Some results on parallel concatenated coding schemes,” *IEEE Trans. Inform. Theory*, vol. 42, pp. 409–428, Mar. 1996.

[86] L. C. Perez, J. Seghers, and D. J. Costello, “A distance spectrum interpretation of turbo codes,” *IEEE Trans. Inform. Theory*, vol. 42, pp. 1698–1708, Nov. 1996.

[87] S. ten Brink, “Convergence behavior of iteratively decoded parallel concatenated codes,” *IEEE Trans. Commun.*, vol. 49, p. 17271737, Oct. 2001.

[88] B. Zhao and M. Valenti, “Distributed turbo coded diversity for the relay channel,” *IEE Electronics Letters*, vol. 39, pp. 786–787, May 2003.

[89] D. Divsalar and F. Pollara, “Multiple turbo codes,” in *Proc., IEEE MILCOM*, pp. 279–285, Nov. 1995.

[90] D. N. Rowitch and L. B. Milstein, “On the performance of hybrid FEC/ARQ systems using rate compatable punctured turbo (RCPT) codes,” *IEEE Trans. Commun.*, vol. 48, pp. 948–959, June 2000.