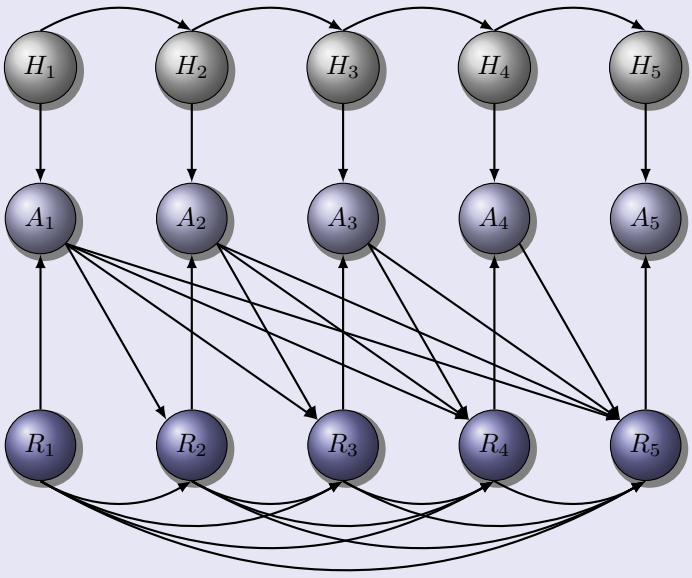
Agile wireless transmission strategies

Agile Wireless Transmission Strategies

Chin Keong Ho



Agile Wireless Transmission Strategies

Chin Keong Ho

Agile Wireless Transmission Strategies

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Technische Universiteit Eindhoven, op gezag van de
Rector Magnificus, prof.dr.ir. C.J. van Duijn, voor een
commissie aangewezen door het College voor
Promoties in het openbaar te verdedigen
op dinsdag 3 maart 2009 om 16.00 uur

door

Chin Keong Ho

geboren te Singapore, Singapore

Dit proefschrift is goedgekeurd door de promotor:

prof.dr.ir. J.P.M.G. Linnartz

Copromotor:
dr.ir. F.M.J. Willems

CIP-DATA LIBRARY TECHNISCHE UNIVERSITEIT EINDHOVEN

Ho, Chin Keong

Agile Wireless Transmission Strategies / by
Chin Keong Ho. -
Eindhoven : Technische Universiteit Eindhoven, 2009. -
Proefschrift. – ISBN: 978-90-386-1519-6
NUR 959
Subject headings: wireless communications \ rate adaptation \ automatic repeat request \ OFDM modulation \ majorization theory \ dynamic programming

Cover design by Chin Keong Ho

Samenstelling van de promotiecommissie:

prof.dr.ir. J.P.M.G. Linnartz, Technische Universiteit Eindhoven, promotor
dr.ir. F.M.J. Willems, Technische Universiteit Eindhoven, copromotor
prof.dr.ir. S.C. (Sem) Borst, Technische Universiteit Eindhoven
prof.dr. Behrouz Farhang-Boroujeny, University of Utah
prof.dr. R.D. van der Mei, Centrum voor Wiskunde en Informatica
dr.ir. Job Oostveen, TNO Information and Communication Technology
dr. Sumei Sun, Institute for Infocomm Research, A*STAR, Singapore

ABSTRACT

Agile Wireless Transmission Strategies

Wireless communications has received much research interest in recent years. In wireless communications, information is conveyed using a wireless channel, from a node acting as transmitter to a node acting as receiver. As there is no need to lay cables, wireless communication systems can be deployed easily. Mobile communications, i.e., wireless communications with portable communication devices, bring further convenience to the end users, by permitting them to establish communications on the go.

The phenomenal success of wireless communications has led to a high demand, and hence scarcity of the limited wireless spectrum. Yet, bandwidth-hungry applications, such as video streaming, that require the transfer of large amounts of data, are becoming popular. The increase in number of users also results in a dense spatial reuse, i.e., a large number of transmissions per unit area. To achieve a high throughput with limited bandwidth in a limited area, it is important to optimize the use of radio resources over time, frequency and space. This is a challenging task as the wireless channel is time-varying in nature – a small change in the environment can cause the received signals to change considerably in amplitude and phase. These signal variations become worse in mobile communications, where the transmitter and receiver typically move. These variations also become worse in multi-user wireless communications, where a possibly varying number of users share the same wireless channel and generate varying amounts of multi-user interference.

The transmitter is, in general, not continuously aware of the instantaneous conditions or state of the channel. However, a limited amount of *channel state information* (CSI) can often be made available by explicit feedback from the receiver, or by listening to the environment. Based on this CSI, transmission parameters (e.g. rate, power, frequency and time of transmission) can be chosen to optimize a measure of system performance (e.g. long-term throughput) via a suitable *adaptation algorithm*. The precise CSI that is used, the set of allowable transmission parameters, and the adaptation algorithm together make up a *transmission strategy*. To optimize the use of the radio resources, this strategy has to be designed properly.

Instead of specifying rigid standards for wireless communications, it is becoming common that baseline standards are specified with hooks for proprietary add-ons and enhancements. This offers the possibility for static and dynamic optimization of the transmission parameters. For example in the recent IEEE 802.11n standards, the

choices of modulation constellation, code rate, packet aggregation, automatic repeat request (ARQ) policy, and multi-input multi-ouput (MIMO) transmission mode are left to the device makers, and the adaptation algorithms for choosing these parameters are left open for future progress and competitive advantages. In this thesis we seek to exploit these degrees of freedom as a basis for attractive transmission strategies. The insights we obtain can also be useful for designing future cognitive radios, in which communication devices are empowered to make informed decisions on how and when to transmit and receive.

In wireless communications, especially in mobile communications, the attractiveness of a transmission strategy is determined largely by its agility. By an agile transmission strategy we mean a strategy that is simultaneously *lean*, *responsive* and *simple*:

1. Lean in feedback: It is essential that the amount of feedback from receiver to transmitter is small, so that the overhead is negligible compared to the improvement in throughput (in number of bits recovered per channel use).
2. Responsive in adaptation: The nodes should be able to react rapidly and adequately to a changing wireless environment.
3. Simple in implementation: The complexity of the adaptation algorithm should be low to enable implementation in the mobile device, where the primary considerations are to have a low battery drain and a small form factor.

The objectives of being simultaneously lean, responsive and simple are often contradictory, making the design of an agile transmission strategy a challenging task. To be lean, only partial CSI can be used in the transmission strategy. To derive an optimal strategy based on this limited knowledge involves more complex operations compared to when the channel state is known exactly, as the uncertainty of the channel state has to be taken into account. As a result, the transmission strategy becomes complex and is *not* simple to implement. Similarly, in being lean, the transmission strategy may not be sufficiently responsive, due to the limited channel knowledge.

Many communication systems have evolved towards packet-switched systems, in which information bits are aggregated into blocks called packets which are separately transported from node to node. To successfully transport each packet, different fundamental aspects of communication across the hostile wireless channel are often solved independently across a hierarchy of communication layers. In this thesis we focus on the lowest two layers: the data link layer (DLL) and the physical (PHY) layer. Information bits to be transmitted first arrive at the DLL. The logical link control (LLC) sublayer, which is the upper sublayer of the DLL, takes note of the information bits that are sent and arranges for retransmission later if necessary. The bits are then passed to the lower sublayer, the medium access control (MAC) sublayer. The MAC sublayer decides when to access the channel so as to reduce inter-user collision. Finally, the PHY layer modulates the bits for transmission. After passing through the wireless channel, the received signal is passed to the PHY layer at the receiver for decoding. The information that the decoding succeeds or fails, and the decoded bits if available, is then passed to the upper layers.

In this thesis, we develop agile transmission strategies for the PHY layer and the MAC

and LLC sublayers to improve bandwidth efficiency and increase user throughput. To this end, we consider packet-by-packet transmit adaptation to ensure responsiveness. We demonstrate that even with lean feedback, a substantial throughput gain can be achieved in hostile wireless channels compared to the case of no feedback. Furthermore, most of this gain can be achieved by using simple adaptation algorithms, instead of using optimal but often highly complex algorithms. To facilitate an effective market introduction, the proposed agile transmission strategies in this thesis fit into established common practices in wireless systems and build upon existing standards. For instance, for adaptation purposes we improve and extend on the use of acknowledgement (ACK) bits, which are already used as feedback in the protocol by the IEEE 802.11 standard.

We start by considering single-carrier communication systems. In Chapter 2, we perform rate adaptation in the LLC sublayer. A lean CSI based on ACK feedbacks is provided by an ARQ scheme. We propose a simple implementation which achieves a significant improvement in throughput compared to no feedback. In Chapter 3, we propose a more advanced ARQ scheme implemented jointly with rate adaptation. Despite its improved performance, the implementation remains simple and the feedback remains lean. Next, multi-carrier communication systems are considered. In Chapter 4, we consider a pre-transformed orthogonal frequency division multiplexing (PT-OFDM) system. We propose an iterative receiver algorithm with a low implementation complexity. The extension of PT-OFDM systems to include ARQ is considered in Chapter 5, in which we propose a simple subcarrier-assignment scheme. In Chapter 6, we consider multi-user communications and include the MAC sublayer. The request-to-send (RTS) and clear-to-send (CTS) mechanism is used to mitigate multi-user interference, and by using a new successive-capture analysis, we perform rate adaptation which uses the RTS/CTS signalling as a lean CSI. Finally, Chapter 7 provides conclusions and recommendations for future research.

Contents

List of Figures

List of Tables

CHAPTER 1

Overview

"The term adaptive communication is perhaps more appropriate than wireless communications."
Randy H. Katz [1]

1.1 Wireless Communications

A wireless communication network consists of multiple communication nodes, acting as transmitters or receivers. To establish wireless communication, a band in the electromagnetic spectrum is reserved and shared by the transmitters and receivers in

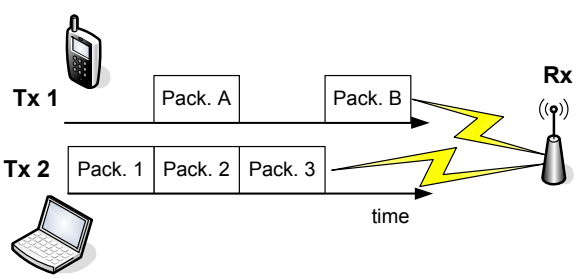


Fig. 1.1: A wireless network consisting of two transmitters (Tx 1 and Tx 2) and one receiver (Rx). Packet switching is employed, sometimes in an uncoordinated manner: Tx 1 sending Packets A, B shares the channel with Tx 2 sending Packets 1, 2, 3.

a spatial neighborhood. An example of a wireless network is illustrated in Fig. 1.1, in which two transmitters Tx 1 and Tx 2 transmit to a receiver Rx.

Traditionally, circuit switching is used to transport data from one end node to another end node, possibly via other intermediate nodes. In circuit switching, each channel between two nodes is reserved for communications over a continuous period of time. It is becoming increasingly common nowadays that data are transported using the internet protocol (IP). In this protocol, information bits are aggregated into blocks called packets, and each packet (rather than each bit) becomes the basic unit of data for transport. In contrast to circuit switching, the IP employs packet switching, where packets are dynamically routed between nodes before they reach their final destinations. Further, each channel in a route is shared among different users, where this sharing is determined by a multiplexing scheme.

Future networks, whether wired or wireless, are converging towards an all-IP network [2], as a result of the ubiquity of the IP. Hence, packet switching plays a prominent role in modern wireless communications. For illustration, Fig. 1.1 shows that Tx 1 is sending Packets A, B, while Tx 2 is sending Packets 1, 2, 3. In wireless networks packet transmissions may not be perfectly coordinated. For instance in Fig. 1.1, we see that Packet A and Packet 2 are transmitted concurrently, which is seen as a collision at Rx.

In wireless communications, the transmitter or receiver is likely to be mobile, especially for personal communication devices such as hand phones and laptops. This form of communications, known as mobile communications, is characterized by faster channel variations over time, as compared to fixed wireless systems where the transmitters and receivers are in fixed positions. In mobile communications, channel tracking can be significantly more difficult. Furthermore, the form factor and battery drain are important design considerations that affect the feasibility and marketability of mobile devices. The complexity of the algorithms implemented in the transmitter and the receiver therefore has to be kept low.

1.1.1 Scarce Spectrum

The electromagnetic spectrum typically used in wireless communications ranges from around tenths of MHz to tens of GHz. Currently, the actively used spectrum starts from frequencies in the order of $10-100$ MHz for radio and TV broadcast, to frequencies in the order of 1 GHz for wireless local-area networks (LANs) and cellular phone services, and then to frequencies in the order of 10 GHz for fixed wireless services. The spectrum is nevertheless limited and its use is strictly governed by regulations. Most portions of the spectrum are reserved for use by licensed nodes, such as the 3G spectrum for cellular phone services. The remaining portions, such as the industrial, scientific and medical (ISM) bands, are unlicensed. Nodes communicating in ISM bands must accept or mitigate the interference generated by other ISM users.

The deployment of wireless communication systems has become increasingly widespread in recent years. This phenomenon fuels the demand for transmission bandwidth. Fur-

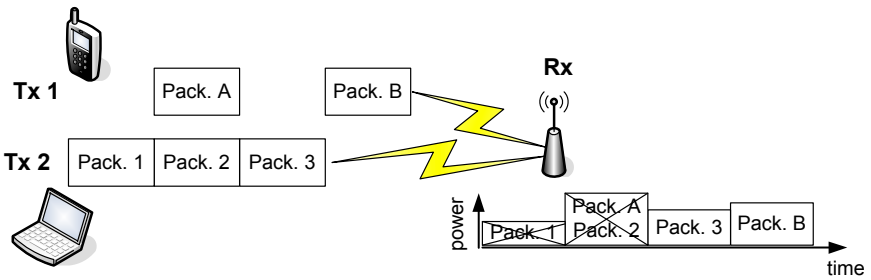


Fig. 1.2: Some challenges in wireless communications: Packet A and Packet 2 are not recovered due to multi-user interference, while Packet 1 is not recovered due to a deep channel fade.

thermore, applications that require the transfer of large amounts of information within a relatively short time, are becoming popular. These applications inherently demand larger bandwidth. As the demand for bandwidth increases, yet with the usable spectrum remaining unchanged, it is becoming increasingly important to optimize the use of every radio resource.

1.1.2 Fast Channel Variations

In an in-building wireless channel, the transmitted signal propagates through the air via multiple paths by reflecting or diffracting from scatterers (e.g. people, walls), before eventually reaching the receiver. When there is a line of sight (LOS) from the transmitter to the receiver, the multipaths include a strong LOS path. The received signal is the combination of these multipath signals. Depending on the phases of the multipath signals, this combination can be constructive or destructive. The power of the received signal can hence be low or high. If the power is low, the signal is said to have experienced a deep channel fade. This can cause a failure in recovering a packet, such as Packet 1 in Fig. 1.2.

Multipath fading is observed regardless of whether there is an LOS path from the transmitter to the receiver, as long as there are at least two paths. The extent of the channel variations is however more significant for non-LOS channels than for LOS channels. The *speed* of the channel variations, on the other hand, depends on the (relative) changes in the environment over time. The speed is usually higher in mobile communications, as the movement of the transmitter or the receiver induces more channel variation. Tracking of the channel becomes more difficult as the extent and speed of variations increase.

1.1.3 Dense Spatial Reuse

The wireless channel is a broadcast medium and is shared by users operating in a common band in a common area or neighborhood. Hence, multi-user interference

occurs when multiple users transmit concurrently. For example, in Fig. 1.2, the transmissions of Packet A and Packet 2 have overlapped, causing a so-called collision in which neither packet is recovered due to the strong mutual interference. If such strong interference occurs frequently, practically no data can be recovered at any receiver.

The number of wireless users has increased rapidly over recent years, and this growth is expected to continue in the near future. Since the spectrum is limited, more users in the same neighborhood are forced to share the same band for wireless communications, resulting in a denser spatial reuse and a higher probability of collision.

1.1.4 The Challenge

Because of the scarcity of the spectrum, we should optimize the use of every transmission opportunity, such that the amount of successfully recovered bits per channel use per unit area is large. Yet, in wireless communications, especially mobile communications, the channel can change from one state to a different state as and when packets are transmitted. Furthermore, the effects of multi-user interference and channel fading have to be mitigated to realize high throughput. In spite of these difficulties, the challenge of providing high throughput has to be met in future-generation communication systems [3].

1.2 Transmission Strategies

1.2.1 Paradigm Shift

To deal with the changes in the wireless channel, the communication systems or networks have to adapt to the environment [1]. Instead of specifying rigid standards for wireless communications, it is becoming common that baseline standards are specified with hooks for proprietary add-ons and enhancements. This offers the possibility for static and dynamic optimization of the transmission parameters. For example in the recent IEEE 802.11n standards, the choices of modulation constellation, code rate, packet aggregation, automatic repeat request (ARQ) policy, and multi-input multi-ouput (MIMO) transmission mode are left to the device makers, and the adaptation algorithms for choosing these parameters are left open for future progress and competitive advantages. In this thesis we seek to exploit these degrees of freedom as a basis for attractive transmission strategies.

Traditionally, much engineering emphasis has been placed on the design of the receiver. The receiver is designed to mitigate the channel fading and interference after the packet is received. Recently, the attention has broadened towards the investigations of "how to transmit". That is, the transmitter is designed to track the channel and to adapt the transmission accordingly *before* the packet is sent. The main focus of this thesis is on transmit adaptation, due to the significant potential performance

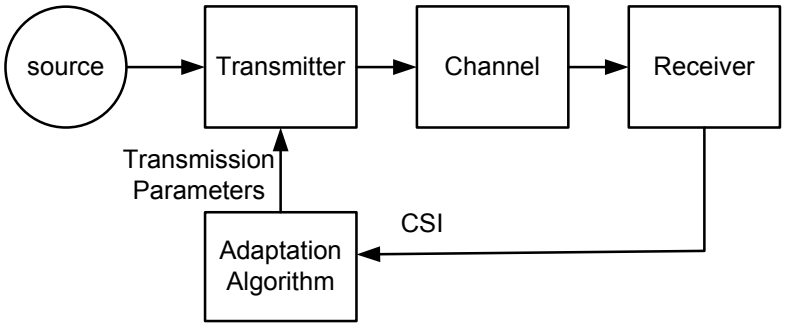


Fig. 1.3: Illustration of a transmission strategy. In general, the channel experiences different channel states. The channel state information (CSI) represents partial information on the channel state that is available at the transmitter. We focus on the case when this CSI is provided by the receiver. The complete design of the transmission strategy involves considerations of the CSI available, of the transmission parameters available, and of the adaptation algorithm.

gain that can be achieved. We focus on transmitting data using packets, since almost all future communication systems will use packet-switched transmission. The insights we obtain can also be useful for designing future cognitive radios [4], in which communication devices are highly empowered to learn and make informed decisions on how to transmit and receive optimally.

1.2.2 Definition of Transmission Strategy

The transmitter is, in general, not continuously aware of the instantaneous conditions or the exact state of the channel. However, *channel state information* (CSI) that partially reflects the channel state can often be made available by explicit feedback from the receiver. Based on this CSI, *transmission parameters* (e.g. rate, power, frequency and time of transmission) can be chosen to optimize a measure of system performance (e.g. long-term throughput) via a suitable *adaptation algorithm*. Fig. 7.2 captures these key elements of transmit adaptation. In short, the CSI acts as an input for the adaptation algorithm to tune the transmission parameters.

We define a *transmission strategy* to consist of the set of CSI available, the set of allowable transmission parameters, and the adaptation algorithm. To optimize the use of the radio resources and overcome the above challenge, the transmission strategy has to be properly designed.

1.2.3 Agility

For wireless communications systems, and in particular for mobile systems, the attractiveness of a transmission strategy is determined largely by its agility. By an agile

transmission strategy we mean a strategy that is simultaneously *lean* in feedback, *responsive* in adaptation and *simple* in implementation:

1. Lean: It is essential that the amount of feedback from receiver to transmitter is small, so that the overhead is negligible compared to the throughput gain obtained by the transmission strategy. Hence, only partial CSI involving a limited amount of feedback is used by the adaptation algorithm.
2. Responsive: The nodes should be able to react rapidly and adequately to a changing wireless environment. In packet-switched communication systems, the packets are designed so that the channel does not change significantly within a packet interval. In this case, the transmission strategy should adapt on a packet-by-packet basis, or whenever the channel is likely to change.
3. Lightweight: The complexity of the adaptation algorithm should be low. This allows the adaptation algorithm to be easily implemented in the mobile device, where the primary considerations are to have a low battery drain and a small form factor.

The objectives of being simultaneously lean, responsive and simple are often contradictory, making the design of an agile transmission strategy a challenging task. To be lean, only partial CSI can be used in the strategy. To derive an optimal strategy based on this limited knowledge involves more complex operations compared to when the channel state is known exactly, as the uncertainty of the channel has to be taken into account. As a result, the transmission strategy becomes complex and is *not* simple. Similarly, in being lean, the transmission strategy may not be sufficiently responsive, due to the limited channel knowledge.

1.3 Layering in Communications

In wireless communications, different fundamental aspects of communication across the hostile wireless channel are often solved independently across a hierarchy of communication layers. The design and implementation of communication systems pertaining to different layers can be efficiently carried out by different teams of engineers. This significant simplification significantly contributes to the widespread deployment, efficient design, and popularity of communication networks.

1.3.1 Open Systems Interconnection (OSI) Model

The open systems interconnection (OSI) reference model defines seven layers [5], as depicted in Fig. 1.4. The layers, from the top to the bottom, are the application, presentation, session, transport, network, data link and physical layers. At the transmitter, data and instructions on how to send the data are passed from each upper layer to the next lower layer. At the receiver, data is passed from each lower layer to the next upper layer to be recovered. Each layer operates independently from other layers, so that peer layers residing at the transmitter and receiver appear to

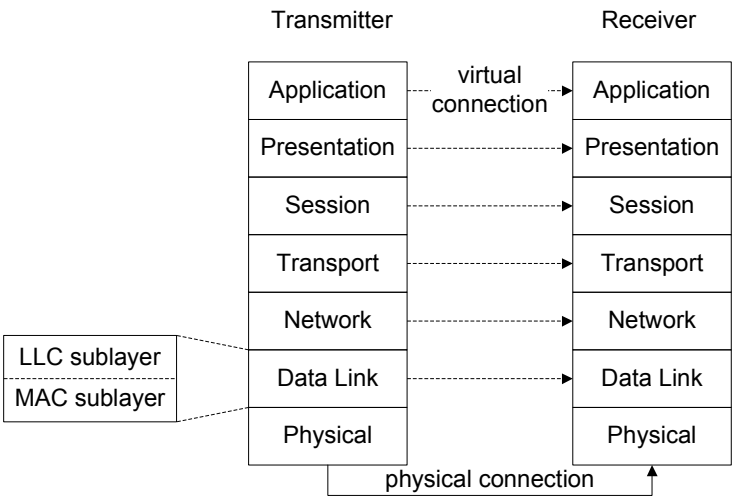


Fig. 1.4: Layering in communications based on the OSI reference model.

communicate directly with each other via a virtual connection, although the physical connection is established only at the physical layer.

In this thesis we focus on the lowest two layers: the data link layer (DLL) and the physical (PHY) layer. The DLL facilitates the transfer of bits from the source to a destination and consists of two sublayers: an upper sublayer, known as the logical link control (LLC), and a lower sublayer, known as the medium access control (MAC). The two sublayers are shown in Fig. 1.4.

The LLC sublayer performs multiplexing and demultiplexing of bits and flow control of packets, including detection and retransmission of packets, if necessary. The MAC sublayer facilitates the sharing of a physical medium by multiple nodes. The PHY layer is the lowest layer. It provides the actual means of modulating the bits as an electrical signal for transmission through the physical medium. In wireless communications, the modulated signal is radiated over a band of frequency in the electromagnetic spectrum.

1.3.2 Automatic Retransmission Request (ARQ)

Retransmissions are orchestrated in the LLC sublayer, by automatic repeat request (ARQ). We elaborate on ARQ due to its significance in this thesis. The concept of ARQ has been introduced since the early 1960s [6,7]. In an ARQ system, each packet is acknowledged when received correctly by sending a positive ACK (PACK) to the transmitter. Otherwise, a negative ACK (NACK) is sent. If the ACK, which can be either a PACK or NACK, is not received by the transmitter within a time limit, the transmitter usually considers that a NACK is received. The transmitter can choose to re-transmit the previous packet for non-delay sensitive data.

In the literature, ARQ schemes are generally classified by considering the capability of the error correction codes used for transmission and retransmission [8]. In the most primitive ARQ scheme, a packet is sent uncoded, without any redundancy for error correction. The receiver requests a retransmission when errors are detected. In the *hybrid ARQ* scheme, the packet is also *encoded* using an error correction code. The receiver first attempts to correct the errors, failing which a retransmission is requested.

A finer classification of the ARQ schemes has been made in the recent literature [8]. In the Type I hybrid ARQ scheme [6], the receiver discards past packets that were received in error and the transmitter encodes each re-transmission independently. In order to improve reliability, the Type II and Type III hybrid ARQ schemes buffer and make use of previous erroneous packets at the receiver for decoding. In Type II hybrid ARQ schemes [7], the re-transmitted packet can be decoded only jointly with past failed packets. In Type III hybrid ARQ schemes, in addition to the possibility of performing joint decoding with past packets, each packet must be also self-decodable, independent of past packets.

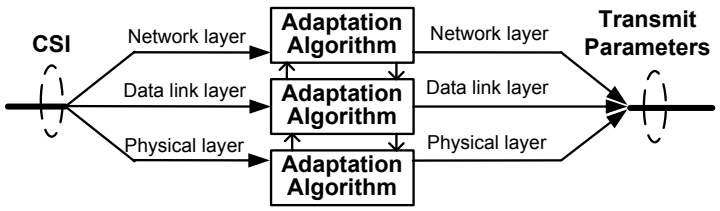
Type I hybrid ARQ schemes are commonly used in current wireless communication systems, such as in wireless LANs [9]. On the other hand, Type II and Type III hybrid ARQ schemes have the higher potential of improving the throughput substantially, by joint decoding with past packets. Thus, Type II and Type III ARQ schemes are investigated in this thesis.

1.3.3 Cross-Layer Adaptations

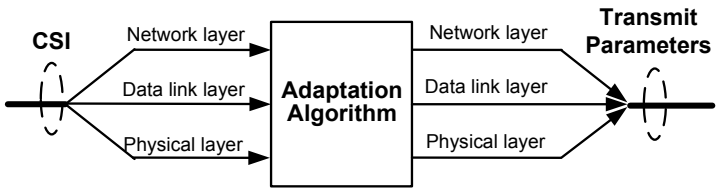
The demand for higher throughput has increased while bandwidth of the wireless channel remains constant. There is a need to reconsider whether designing and optimizing each communication layer independently is limiting the potential of communication systems and networks. Recent investigations suggest that the new paradigm of *cross-layer adaptation*, known also as cross-layer optimization/control/design, indeed has the potential to achieve significant performance improvement which has so far been restricted by traditional layered design [10–12].

In cross-layer adaptations, information conventionally restricted to single layers is shared among multiple layers. Furthermore, this information can be used to perform joint optimization across several layers over time. For illustration, some communication layers are shown in Fig. 1.5. As shown in Fig. 1.5(a), different layers traditionally use different types of CSIs for adaptation. Optimizations carried out in the adaptation algorithms are carried out independently for each layer, and the results may be passed to other layers by interactions with their corresponding upper and lower layers. In cross-layer adaptations, the CSI can be shared across layers, as shown in Fig. 1.5(b). Moreover, a single adaptation algorithm can perform joint optimization across layers. Clearly, with less restrictions, cross-layer adaptations can achieve further performance gains.

The joint design of the transmission strategy across all layers is prohibitively complex.



(a) In layered communications, each layer uses its own CSI and performs adaptation for its own layer. Information across layers is shared via strictly defined interfaces.



(b) In cross-layer adaptation, the CSI is shared and joint adaptation is performed across layers.

Fig. 1.5: The transmission strategy, which comprises of the set of CSI, the adaptation algorithms and the set transmission parameters, can be designed to operate (a) independently in each layer or (b) across layers. For illustration we have shown only the network layer, the data link layer and the physical layer.

This high design complexity in fact motivates the introduction of the concept of layering in the first place [5]. Current research efforts in cross-layer adaptations have thus focused on pragmatic designs that operate on a local scale. That is, cross-layer adaptations are restricted to a few layers, and local optimization, rather than global optimization, is performed across these layers. Cross-layer adaptations have been categorized into the following four major categories [13].

- *Creation of new interfaces between non-adjacent layers*: information that is not conventionally available is passed from an upper layer directly to a lower layer, or vice versa. Some intermediate layers may be bypassed.
- *Merging of adjacent layers*: all information available in the merged layers are shared, so effectively there is only one super-layer.
- *Design coupling*: a *design layer* (such as the MAC sublayer) is re-designed to exploit the improved capability of another *fixed layer* (such as the PHY).
- *Vertical calibration*: parameters across layers are jointly optimized. This optimization can be performed offline during system design, or in real-time during actual operation of the system.

The work in this dissertation can be classified according to some of these categories, as follows. Chapter 3, which builds on Chapter 2, considers design coupling. Specifically, the DLC layer (the design layer) improves the throughput performance by exploiting the improved capability of the PHY layer (the fixed layer), in which more advanced

codes are designed. Only a lookup table is needed for implementation, resulting in a simple adaptation algorithm. By building on materials in Chapter 4, Chapter 5 considers (partial) merging of adjacent layers in the DLL and PHY layer, by sharing information in both layers. We show that the sharing of some new but lean CSI from the PHY layer already allows the DLC layer to achieve a higher throughput. Finally, Chapter 6 considers the interactions of DLL and PHY layers and creates some partial merging of both layers, which is a form of cross-layer adaptation.

Cross-layer adaptation promises to bring about significant performance gain in future networks, but practical constraints and implementations may nullify such gain. A cautionary perspective has been taken in [14]. Further, issues on tradeoffs such as the amount and type of CSI used for cross-layer adaptation, the design of practical adaptation algorithm, the throughput improvement, etc., are not yet well investigated. In particular, the design of agile transmission strategies for cross-layer adaptation as shown in Fig. 1.5(b) is a challenging ambition.

Although we consider cross-layer designs, in contrast to other literature [11, 12] our focus is on the tradeoffs involved in achieving agile transmission strategies. To this end, we consider cross-layer adaptations across the physical layer and data link layer. Since we employ lean feedback, the channel is not known exactly, and so we employ ARQ techniques to recover erroneous transmissions. In [12], a general model of cross-layer adaptation is considered, under the key assumption that the CSI is known exactly and thus error-free transmission is possible.

1.4 Performance Measures

Various measures can be used to assess the performance of communication systems or networks. To a hardware engineer, relevant measures would include the form factor, hardware complexity, architecture reusability, scalability and power consumption. To a system engineer, key measures include the throughput, delay, peak-to-average power ratio, maximum and average number of users that can be supported, maximum and average data rate, and ease of system deployment. The communication engineer's main interest would usually be the bit error rates (BER) or packet-error rates (PER) for a given signal-to-noise (SNR), subject to some constraints on the above measures. The information theorist derives capacity for a general theoretical channel model, which gives the maximum information rate achievable. A constructive proof for the capacity provides guidelines for the design of coding and communication schemes, and hence illuminates the path of implementing practical systems.

All these measures are important in their own right, but a balanced view should be adopted so that not one measure dominates to such an extent that other measures suffer greatly. In addition, the act of balancing resources depends on the communication system that we are interested in. Hence, a top-down approach of viewing a communication system, starting from the environment and application of the system, to the appropriate measures to be used, and finally to the architecture and implementation,

would likely bring about a cost effective solution.

Recently, a top-down approach has been used in the standardization activities of IEEE 802.11n [15] to develop a high-throughput wireless LAN system. This results in a tight integration of the DLL and PHY layers. Consequently, the throughput as measured at the MAC sublayer is proposed as the primary performance measure. This throughput is the effective average rate transmitted from the source to the destination taking all MAC protocol and retransmission overheads into account.

In this dissertation, we use the throughput as the main performance measure. When communicating in a single-user channel, the channel access is treated as granted to the user. Hence, the throughput is evaluated up to the DLL. When communicating in a multi-user channel, the throughput is evaluated by taking into account the MAC sublayer. The throughput as a metric takes into account both the effects of multi-path fading (since packets have to be retransmitted in deep fades) and also multi-user interference when applicable (since MAC overheads such as collisions are taken into account). In addition, throughput allows a meaningful exploration of packet-by-packet adaptation, whose effect cannot be adequately captured with standard BER and PER measures, and yet is simple enough to characterize.

Further, we frequently focus on the maximum throughput, to provide upper bounds of throughput realized in practice. Other quality of service (QoS) measures that need to be taken into account, such as delay, can be included as optimization constraints.

1.5 Structure of Dissertation

The two lowest layers, i.e., the data link layer (consisting of the MAC and LLC sublayers) and the PHY layer, are closest to the wireless medium, and hence they provide the most direct way to counteract the time variations of the wireless channels. As such, we focus on these layers in this dissertation.

The main chapters of this dissertation are Chapters 2, 3, 4, 5, 6. They are structured as shown in Table 1.1, depending on whether the wireless channel is single-user or multi-user, on whether the communication system in the PHY layer is single-carrier or multi-carrier, and finally on how the data link layer is implemented.

Chapter	Channel	Physical Layer	Data Link Layer
2	multi-user	single carrier	Type I hybrid ARQ (LLC sublayer)
3	single-user	single carrier	Type II hybrid ARQ (LLC sublayer)
4, 5	single-user	multi-carrier	Type II hybrid ARQ (LLC sublayer)
6	multi-user	single carrier	RTS/CTS protocol (MAC sublayer)

Table 1.1: Structure of dissertation.

We consider both single-user channels and multi-user channels in this dissertation. In Chapters 3–5, we focus on single-user channels. In single-user channels, the transmis-

sion of every user is centrally coordinated to limit the spatial, spectral and temporal extent of the interference that it generates to other users. In Chapters 2, 6, we consider multi-user channels. In multi-user channels, each user may see multi-user interference due to a lack of a central coordination. To limit the interference arising from other users to oneself, and also to limit one's interference to other users, a distributed channel access protocol may be employed.

We consider single-carrier communication systems in Chapters 2, 3, 6 and multi-carrier systems in Chapters 4, 5. By multi-carrier systems, we mean that data are transmitted over multiple subcarriers, for instance in an orthogonal frequency division multiplexing (OFDM) system. In contrast, in single-carrier systems, all data are transmitted only over a single carrier. Although we consider single-carrier communication systems in Chapters 2, 3, 6 for simplicity, these chapters provide a framework from which multi-carrier systems can be investigated.

Finally, the dissertation can be structured according to how the LLC sublayer or MAC sublayer is implemented. In Chapters 2–5, we consider single-user channels, so the MAC sublayer is irrelevant. We use a Type I hybrid ARQ scheme in Chapter 2 and a Type II hybrid ARQ scheme in Chapters 3, 4, 5. Next, we focus on the MAC sublayer in Chapter 6. Retransmission is not carried out here, which makes the LLC sublayer irrelevant, but the RTS/CTS protocol is used to gain channel access. The RTS/CTS protocol is activated on demand when data arrives to reserve the channel over a spatial region, which ensures an interference-free transmission.

We now give a summary for each of the main chapters, and highlight the agility of the transmission strategies employed.

1.5.1 Chapter 2: Rate Adaptation using ACK Feedback

1.5.1.1 Background and Problem Formulation

Due to user mobility or changes of the environment, the wireless channel often experiences fading which causes the received signal power to vary over time. Further, varying interference, due to transmissions from other users sharing the same band, or radiation from, for instance, microwave ovens, alters the channel in a random manner.

To ensure packet recovery in a time-varying channel, the rate of the packet needs to be matched to the instantaneous channel condition. This process is known as *rate adaptation* [16]. The CSI required for rate adaptation at the transmitter can be provided by the receiver. Although a more informative feedback leads to a higher throughput, in practice the availability of feedback is restricted by the wireless scenario and the communication system. Rate adaptation that requires additional feedback beyond the standard ACK feedback [17, 18] is not compliant with many legacy devices, such as those based on the IEEE 802.11a/b/g standards. Moreover, channel reciprocity is not always valid, i.e., the return channel may not behave identically as the forward channel, e.g. in frequency division duplex systems. This could limit the effectiveness of rate adaptation schemes which exploit channel reciprocity, such as [16, 19, 20].

Full compatibility with any basic ARQ system, even in systems without channel reciprocity, can be ensured if only the history of ACKs is used as CSI [21–23]. In fact, past rates used for previous transmissions can be stored in memory which also serve as CSI [24, 25]; no additional feedback is required. In [24], a common rate is used for transmission by several packets in a frame. The CSI consists of this common rate and the ACKs within the previous frame, but earlier rates and ACKs are discarded and not used as CSI. In [25], *all* past ACKs and rates are used as CSI; we refer to this form of CSI as *ACK-rate CSI*.

The problem of maximizing throughput by rate adaptation using the ACK-rate CSI is a PSPACE-complete problem [25, 26], which is at least as hard as an NP-complete problem. This means that optimal rate schemes cannot be implemented practically. Moreover, it is not clear the maximum achievable throughput can be quantified. As such, an upper bound on the maximum achievable throughput is obtained in [25]. However, this upper bound may not be sufficiently tight in some channels, and should ideally be improved upon. A tight upper bound is desirable since it provides an accurate indication of how close a rate adaptation scheme performs with respect to the optimal one. On the other hand, if the gap in performance is found to be small, using a more complicated scheme would then not be worthwhile.

Moreover, since the optimal rate adaptation cannot be implemented practically, rate adaptation schemes based on heuristics have to be devised. Although several rate adaptation schemes are proposed in [25], the complexity of these schemes can increase quickly as the number of possible rates becomes large. It is desirable that simple algorithms for rate adaptation are developed, while all past ACKs and rates are exploited as (lean) CSI, so as to lead to agile transmission strategies that are more appealing for practical implementations.

Although collisions due to multi-user interference occur frequently in practice, collisions have not been considered in [25, 26]. As such, an overly conservative rate adaptation scheme may result, because an NACK caused by a collision may be wrongly perceived to be caused by a deep channel fade.

1.5.1.2 Contributions

In this chapter, to match variations in channel conditions, we employ rate adaptation and seek to maximize the throughput averaged over an infinite time horizon. To limit the feedback, ACK-rate CSI is employed. The ACK feedback is used for rate adaptation in IEEE 802.11a/b/g systems. One of the important challenges in wireless system standardization is to keep the amount of channel feedback small. The use of a one-bit feedback (via ACK) represents the extreme case of limited feedback and is thus useful as benchmark for future schemes or other existing schemes that require more feedback.

Our contributions are as follows. Firstly, we study the effects of multi-user interference on rate adaptations, by modeling collisions in the FSMC. Secondly, we establish two new upper bounds that are tighter than currently known ones. To obtain these

upper bounds, we let the transmitter receive a CSI that is more informative than ACK-rate CSI. Thirdly, we propose practical near-optimum rate adaptation schemes. To reduce the complexity of real-time implementation, we consider the pragmatic approach of maximizing over a finite time horizon. We propose the *particle-filter-based rate adaptation* (PRA), which employs the particle filter [27] for rate adaptation. The PRA has a complexity that is largely independent of the number of rates used. This allows us to use a large number of rates when we explore the potential of rate adaptation.

Numerical studies show that the throughput performance drops drastically if collisions are not properly accounted for. Moreover, the proposed PRA outperforms conventional schemes and performs within one dB of signal-to-noise ratio (SNR) to the proposed upper bounds for a slowly changing channel, even in the presence of collisions.

1.5.2 Chapter 3: IRID ARQ Coding Scheme

1.5.2.1 Background and Problem Formulation

In Chapter 2, Type I hybrid ARQ schemes are considered. Although failed transmissions contain some information about the data symbols, they are discarded and not exploited in these schemes. In Chapter 3, Type II hybrid ARQ schemes are considered instead. In these schemes, failed transmissions are saved in a buffer at the receiver. These failed transmissions are then used jointly with the present retransmission for decoding the data symbols. Since joint decoding improves the probability of decoding the data correctly, the throughput is also improved.

The redundancy bits in the retransmissions can be sent incrementally in the form of so-called *incremental redundancy* (IR) bits, which are usually significantly smaller in number than the bits sent in the failed transmission. Typically, IR is continually sent in small blocks until the data symbols are finally decoded successfully and a positive ACK (PACK) is received. At the point when the PACK is received, a just sufficient amount of redundancy has been used to successfully decode the data. Hence, IR enables the efficient use of channel resources. A well-known IR code is the rate compatible punctured convolutional code (RCPC) [28]. More advanced capacity-approaching IR codes based on turbo codes [29] and low-density parity-check codes [30] have also been designed.

In some communication systems, such as in a time-division multiple access (TDMA) system, fixed blocks of channel resources called slots are reserved for transmissions for each user. Hence, every packet is transmitted in one of these slots. Typically, the *entire* slot is used to send redundancy bits in retransmissions. However, for some channel scenarios, only slightly more redundancy bits are needed for successful decoding of the data. In such cases, more channel resources than necessary have been used for transmitting redundancy bits. The excess channel resources could have been

used instead to transmit new information bits that are previously unsent. In other words, the channel resources have not been utilized efficiently.

1.5.2.2 Contributions

To improve the utilization of channel resources in slot-based systems, we send IR in retransmission packets using Type II hybrid ARQ schemes. We propose to send incremental data (ID), i.e., information bits not previously sent and may be appropriately coded, using the remaining channel resources that are not used for sending IR. We call this ARQ scheme the incremental-redundancy incremental-data coding (IRIDC) scheme. This is appropriate when there are always sufficient number of bits waiting to be sent at the source, such as in applications like multimedia streaming. To achieve an agile transmission strategy, we seek to keep the implementation of the code simple and the feedback lean.

For flexibility in implementation, we design a code that allows an arbitrary number of IR bits and incremental data bits to be sent in each packet. Then, we propose to encode all bits (including the previous ones that failed to be recovered and new ones to be sent) using a single *effective rate*. Hence, this code requires only a single-bit ACK feedback to be fed back, resulting in a lean feedback. Moreover, we propose to employ time multiplexing to send the IR bits and coded incremental data bits in retransmissions, resulting in a simple implementation.

Numerical results obtained in Rayleigh fading channels reveal the effectiveness of the proposed code. Though this code is simple to implement and requires only a lean feedback, it achieves close to the maximum possible throughput when the channel is exactly known at the transmitter. In particular, by optimally choosing the effective rates over four packets given only a (causal) ACK feedback for each transmitted packet, our solution achieves a throughput that is less than 1 bit/symbol short of this maximum throughput, for any SNR less than 30 dB.

1.5.3 Chapter 4: Iterative Subcarrier Reconstruction in OFDM Systems

1.5.3.1 Background and Problem Formulation

Wireless channels typically are frequency selective, i.e., the channel transfer function varies over frequency, as a consequence of the presence of multiple propagation paths. Orthogonal frequency division multiplexing (OFDM) is a digital modulation technique that is highly suitable for transmission over frequency-selective channels [31]. A number of wireless standards such as IEEE 802.11a [9], 802.11n [15], WiMax [32] and long term 3G evolution [33] have adopted OFDM-based solutions for physical-layer transmission.

In an OFDM system, data symbols are transmitted over multiple subcarriers, where each subcarrier usually carries one stream of data symbols. Since the transmission bandwidth is divided among the subcarriers, each subcarrier occupies a smaller bandwidth and effectively becomes a flat-fading channel, i.e., the subcarrier channel is not frequency-selective. Furthermore, in OFDM a so-called cyclic prefix is inserted before every OFDM transmission. The cyclic prefix is a repetition of the last portion of the OFDM transmission and it allows the inter-symbol interference to be isolated and contained within each OFDM symbol. At the receiver, a detector is used to recover the data. During detection, equalization is carried out to remove or reduce signal distortions introduced by the channel. The use of a cyclic prefix allows a low-complexity detector to recover these data. In particular, a detector that involves only linear operations for the equalization, known as a *linear detector*, is often used in OFDM receivers. On the other hand, non-linear detectors, such as maximum-likelihood (ML) detectors that employ exhaustive searches, typically require substantially higher complexity than linear detectors. Thus linear detectors are preferred because of their simplicity.

In OFDM systems the data symbols sent on some carriers can be in deep fades and thus fail to be recovered at the receiver. This has motivated the proposal of more robust transmission schemes in which the information is spread across all the subcarriers by some pre-transformation (PT) matrix, referred to here in general as PT-OFDM.

In the literature, the transform used for the PT-OFDM system has been optimized for linear detectors [34] and for ML detectors [35, 36]. Although linear detectors are simpler to implement, the BERs achieved are worse than those that use ML detectors. On the other hand, although ML detectors offer better BER performance, their high complexity may not be suitable for implementations in systems that employ a large number of subcarriers. Iterative detectors, in which a mix of linear and non-linear equalization techniques are carried out iteratively, offer a good tradeoff. In each iteration, the complexity is much lower than using ML-like non-linear detectors, yet the performance can improve with increasing number of iterations. Overall, iterative detectors can hence give good performance at acceptable complexity. In [37, 38], an iterative detector based on parallel interference cancelation (PIC) is proposed for multi-user detection. In PT-OFDM, the symbols transmitted in each subcarrier can be treated as being transmitted by a user, and hence PIC can be applied for detection for PT-OFDM. However, the complexity of PIC becomes high as the number of subcarriers becomes large. Hence, a detector that has good BER performance with lower complexity than PIC is needed.

1.5.3.2 Contributions

Since wireless channels are typically frequency-selective, in this chapter we focus on PT-OFDM systems. We seek to overcome the above challenge of designing simple detectors with performance that is comparable to PIC. To this end, we choose to design an iterative detector since it generally provides a good tradeoff between complexity

and performance, but we further exploit differences in channel powers at different subcarriers to develop a simple detector.

Specifically, to keep complexity low, we employ a linear detector in each iteration of our proposed iterative detector. For frequency-selective channels, the use of the linear detector can enhance noise at weak subcarriers, i.e., those in deep channel fades. Consequently, the overall signal quality can be significantly degraded by these weak subcarriers. Hence, we first aim to reduce the noise enhancement at the weakest subcarrier, by *reconstructing* the signal received there. That is, we replace that received signal with an estimated value. The estimate is obtained based on the tentatively-detected symbols in the previous iteration. At moderate SNR, this estimate usually introduces a smaller amount of propagation error (from wrong decisions made for the detected symbols) as compared to the noise enhancement, hence the overall noise is reduced. The reconstruction is performed iteratively for the next weakest subcarrier, and so on.

We provide an analysis on the optimization of the transform coefficients and reconstruction method so as to maximize the minimum SNR. We analyze the error performance if there were no error propagation and show that the diversity is increased by one for every additional reconstruction performed.

Numerical results show that the proposed detector based on iterative reconstruction has lower complexity than ML detector and PIC. Moreover, it performs significantly better than linear detectors and also better than PIC, and close to the ML detector over a wide range of SNRs.

1.5.4 Chapter 5: ARQ by Subcarrier Assignment

1.5.4.1 Background and Problem Formulation

Continuing from Chapter 4, we also consider PT-OFDM systems in Chapter 5. In PT-OFDM or OFDM systems, an ARQ mechanism is commonly used to ensure the successful delivery of data symbols. The ARQ mechanism triggers the transmitter to retransmit data symbols that are not successfully delivered, upon a NACK fed back from the receiver. Typically, the data symbol originally transmitted on an *original subcarrier* is repeated on a different *ARQ subcarrier* in the retransmission. We say that the ARQ subcarrier is *assigned* to the original subcarrier.

In the literature, fixed assignments that do not exploit any channel state information (CSI) have been considered [39–41]. In these works, the ARQ subcarriers are cyclicly-shifted versions of the original subcarriers. If the cyclic shift is larger than the coherence bandwidth, then the failed data symbols are re-transmitted on subcarriers that experience independent channel fades. This assignment exploits frequency-domain diversity and clearly offers an improvement over a simple scheme in which the failed symbols are simply repeated on the same subcarriers.

In some current and future standards, such as IEEE 802.11n [15] and 3G standards

[33], some form of CSI is available via a feedback channel. Exploitation of this CSI for subcarrier assignment can be expected to achieve better performance than fixed-assignment schemes, such as based on cyclic assignments. In cyclic assignments, for some channel realizations data on a weak original subcarrier can still be retransmitted on a weak ARQ subcarrier, resulting in poor performance. To avoid this problem, it would be desirable to make the assignment adaptive by exploiting the CSI. Moreover, the CSI required should be lean, and the algorithm used to derive the assignment should be simple, so as to obtain an agile transmission strategy suitable for practical implementations.

1.5.4.2 Contributions

As another step towards designing agile transmission strategies, we address the challenge of exploiting lean CSI for subcarrier assignments that are simple to implement. To this end, we consider adaptive subcarrier assignments that depend on channel realizations, specifically in the form of signal-to-noise ratios (SNRs) of the original and ARQ subcarriers. In contrast, previous work in the literature considers fixed subcarrier assignments that did not depend on channel realizations.

Two ARQ subcarrier assignment (ARQ-SA) problems are formulated. The first problem is known as single ARQ-SA, where at most one ARQ subcarrier is assigned to any original subcarrier. We propose the *oppositely-ordered assignment* to solve the single ARQ-SA problem, in which the ARQ subcarrier with the mth largest SNR is assigned to the original subcarrier with the mth smallest SNR. By using the theory of majorization [42], we prove that the oppositely-ordered assignment is optimal in terms of maximizing any utility function that is Schur concave. We show that many typical utility functions are indeed Schur concave.

The second problem is known as multiple ARQ-SA, where any number of ARQ subcarriers can be assigned to any original subcarrier. For the multiple ARQ-SA problem, we show that this assignment problem is NP-hard. We thus propose a sub-optimum subcarrier assignment solution built on top of the oppositely-ordered assignment. Although sub-optimal, this assignment applies to more general scenarios.

Both proposed assignments are simple to implement. To illustrate this, let us consider an assignment for M subcarriers. An exhaustive search for the optimal assignment would require a complexity of $O(M^M)$ for the single and multiple ARQ-SA problems. This complexity becomes prohibitive when M is large, such as in the IEEE 802.16 standards [32] where up to 2048 subcarriers are used. On the other hand, the optimal oppositely-ordered assignment for the single ARQ-SA problem has a complexity of $O(M \log M)$, while the heuristic assignment for the multiple ARQ-SA problem has a complexity of $O(M^2 \log M)$.

To reduce the overhead of signaling the chosen assignment, we propose to group G neighboring subcarriers and perform assignments on these groups. Our analysis indicates that the saving on the overhead is significant even for small G. As such, the CSI can be made lean even by increasing G slightly, at negligible performance loss.

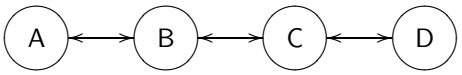


Fig. 1.6: A linear arrangement of nodes where carrier sensing and interference is effective only over neighboring nodes.

Numerical results indicate that substantial throughput improvement can be achieved by the proposed assignments, compared to fixed assignments in the literature that do not exploit CSI. Further, the use of grouping reduces the signaling overhead significantly, yet with little throughput loss.

1.5.5 Chapter 6: Successive-Capture Analysis of RTS/CTS

1.5.5.1 Background and Problem Formulation

In packet-switching, packets can arrive randomly, and hence the number of nodes that wish to share the wireless channel is variable and usually difficult to determine in advance. To support such a variable number of active nodes, a random access MAC protocol is often employed. Well-established random access protocols for the infrastructure networks include ALOHA [43] and physical carrier sensing [44].

In the ALOHA protocol, a node transmits whenever new data arrives, or when this needs to be retransmitted because of a missing acknowledgement (ACK). In the slotted ALOHA variant, transmissions are also globally synchronized. Although simple to implement, ALOHA suffers from excessive interference due to conflicting transmissions when the network load is high [45]. Collisions can be mitigated by physical carrier sensing, where a node is inhibited to transmit if, prior to transmission, it detects signal power from ongoing transmissions. However, physical carrier sensing suffers from the well-known *hidden-node* and *exposed-node* problems.

For illustration, let us consider the scenario where there are four nodes A, B, C, D arranged in a straight line, as shown in Fig. 1.6 (see next page). Carrier sensing is used for random access. We assume that carrier sensing is effective only over neighboring nodes. That is, only transmissions of adjacent nodes are detected by carrier sensing. We also assume that interferers are effective only over neighboring nodes. That is, only transmissions of adjacent nodes can interfere and cause a reception to fail. The hidden node and exposed node problems can be explained as follows.

- Suppose that A transmits to B. However, C cannot sense the power of A. If C (acting as the hidden node) transmits to D, then it interferes with B. Hence, carrier sensing results in the hidden-node problem, which may cause the transmission of A to B to fail unnecessarily.
- Suppose that B transmits to A. We note that C (acting as the exposed node) is inhibited from sending, even though it is too far to A from interfering. Hence, carrier sensing results in the exposed-node problem, which may cause the transmission of B to A to be inhibited unnecessarily.

The request-to-send/clear-to-send (RTS/CTS) protocol, known also as virtual carrier sensing can solve the hidden-node and exposed-node problems [46,47]. The RTS/CTS protocol was first proposed as the MACA protocol [46] where physical carrier sensing is replaced by virtual carrier sensing: the source first informs the destination of its intention to exchange data by issuing an RTS packet, and the destination confirms this with a CTS packet, after which the source sends the DATA packet. All other nodes (including hidden nodes) that recover the RTS or CTS packet are inhibited from transmitting during some specified time interval, to facilitate a successful RTS-CTS-DATA cycle. This *RTS/CTS cycle* can be extended further by an ACK packet from the destination to reduce the delay caused by erroneous cycles at the transport layer [47]. The RTS/CTS protocol can be optionally used in current IEEE 802.11 [48] systems, which is implemented in the distributed co-ordination function (DCF).

Analysis of a simplified or approximate RTS/CTS model provides insights that complement simulation results, such as in [49, 50]. In the pioneering analysis by Bianchi [51], a packet was assumed to be recovered if and only if no other concurrent transmission occurs. This model can be improved by considering capture, depending on path losses and fading of all links, including interference propagation paths. Linnartz [52, 53] considers the capture effect in Rayleigh fading channels and formalizes the analysis using the Laplace transforms of probability density functions (pdfs) of the joint received interference power. The RTS/CTS protocol has been analyzed with capture effect under Rayleigh fading in [54]. In addition, Kim and Lee [55] consider both Rayleigh and shadow fading channels.

So far in the literature [51, 54, 55], the RTS/CTS protocol has been modeled by assuming that if the RTS packet is recovered by the destination, then the CTS, DATA and ACK packets are always recovered too. This becomes optimistic for environments with dense (or even contiguous) spatial reuse. In fact, the boundaries of the inhibited area are fuzzy and harmful transmissions from these fringes are likely. Thus, a better understanding of the effects of the wireless channels on the RTS/CTS protocol is called for. Furthermore, the probability of successfully recovering a packet should be quantified as a function of the rate of the DATA, which then allows the optimum rate used for transmission in the the RTS/CTS protocol to be determined.

1.5.5.2 Contributions

To better understand the effects of the wireless channel on the RTS/CTS protocol, we drop the conventional assumption that the CTS, DATA and ACK packets are always recovered if the RTS packet is recovered. Instead, we consider more realistic, statistically dependent *capture probabilities*, i.e., the probabilities that packets are successfully recovered, as the protocol progresses. We call this successive-capture analysis. In particular, in this analysis it is possible for packets in the later phases of the protocol to fail even if the earlier phases have succeeded.

The successive-capture analysis allows us to give a probabilistic description of how the interfering traffic is reduced by the RTS and CTS inhibitions. We observe that the

RTS/CTS protocol makes an efficient *spatial* reservation of the channel, by inhibiting only nodes that can potentially harm the reception of packets. This demonstrates that the severity of the hidden-node and exposed-node problems has been limited by the RTS/CTS protocol. Moreover, we confirm (and quantify) that for moderately large messages, nodes can achieve a significantly higher throughput when using the RTS/CTS protocol, compared to the ALOHA protocol.

From our investigations we further obtain the optimum rate for data transmission for given RTS and CTS rates. Such rate optimization was not resolved in the literature previously, possibly because without our newly developed successive-capture analysis no meaningful results can be derived.

1.6 Publications by the Author

1.6.1 Journals

1. "Iterative detection for pretransformed OFDM by subcarrier reconstruction,"
   Chin Keong Ho, Zhongding Lei, Sumei Sun, and Wu Yan, *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2842–2854, Aug. 2005.
   See Chapter 4.
2. "Successive-capture analysis of RTS/CTS in ad-hoc networks,"
   Chin Keong Ho and Jean-Paul M. G. Linnartz, *IEEE Trans. Wireless Commun.*, vol. 7, no. 1, pp. 213–223, Jan. 2008.
   See Chapter 6.
3. "ARQ by subcarrier assignment for OFDM-based systems,"
   Chin Keong Ho, Hongming Yang, Ashish Pandharipande, and Jan W. M. Bergmans, *IEEE Trans. Signal Process.*, vol. 56, no. 12, pp. 6003–6016, Dec. 2008.
   See Chapter 5.
4. "Rate adaptation using ACK feedback in finite-state Markov channels with collisions,"
   Chin Keong Ho, Job Oostveen, and Jean-Paul M. G. Linnartz, *IEEE Trans. Wireless Commun.*, accepted, 2008.
   See Chapter 2.

1.6.2 Conference Proceedings

1. "Performance analysis of iterative detection for pre-transformed OFDM,"
   Chin Keong Ho, Zhongding Lei, Sumei Sun, and Yan Wu, *Proc. 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 2, Barcelona, Spain, Sep. 2004, pp. 1332–1336.
   See Chapter 4.
2. "Maximizing throughput of packet switched wireless communication systems,"
   Chin Keong Ho, Frans Willems, and Job Oostveen, *Proc. 43th Allerton Conference*

on Communication, Control, and Computing, Monticello, IL, Sep. 2005.
See Chapter 3.

3. "Rate adaptation in time varying channels using acknowledgement feedbacks,"
Chin Keong Ho and Job Oostveen, *Proc. 63rd IEEE Vehicular Technology Conference*, vol. 4, Melbourne, Australia, May 2006, pp. 1683– 1687.
See Chapter 2.
4. "Calculation of the spatial reservation area for the RTS/CTS multiple access scheme,"
Chin Keong Ho and Jean-Paul M. G. Linnartz, *Proc. 27th Symposium on Information Theory in the Benelux*, Noordwijk, The Netherlands, Jun. 2006, pp. 181–188.
See Chapter 6.
5. "Analysis of the RTS/CTS multiple access scheme with capture effect,"
Chin Keong Ho and Jean-Paul M. G. Linnartz, *Proc. 17th IEEE Personal, Indoor and Mobile Radio Communications*, Helsinki, Finland, Sep. 2006.
See Chapter 6.
6. "ARQ by subcarrier assignment for OFDM-based systems,"
Chin Keong Ho, Hongming Yang, Ashish Pandharipande, and Jan W. M. Bergmans, *Proc. 41st Annual Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2007, pp. 1723–1727.
See Chapter 5.
7. "Rate adaptation using acknowledgement feedback: throughput upper bounds,"
Chin Keong Ho, Job Oostveen, and Jean-Paul M. G. Linnartz, *Proc. IEEE Global Communications Conference*, New Orleans, LA, Dec. 2008.
See Chapter 2.

1.6.3 Patent Applications

1. Chin Keong Ho, Job Oostveen, and Frans Willems, "Transmission rate adaptation with incremental redundancy", WO/2007/036848, 5 Apr. 2007.
2. Chin Keong Ho, Job Oostveen, and Frans Willems, "Method, apparatus and system for error detection and selective retransmission", WO/2007/036855, 5 Apr. 2007.
3. Chin Keong Ho, Hongming Yang, and Ashish Pandharipande, "Subcarrier assignment for re-transmissions", filing in progress, Oct. 2007.

CHAPTER 2

RATE ADAPTATION USING ACK FEEDBACK

In this chapter[1], we investigate packet-by-packet rate adaptation so as to maximize the throughput. We consider a finite-state Markov channel (FSMC) with collisions, which models channel fading and collisions due to multi-user interference. To limit the amount of feedback data, we only use past packet acknowledgements (ACKs) and past rates as channel state information. The maximum achievable throughput is computationally prohibitive to determine, thus we employ a two-pronged approach. Firstly, we derive new upper bounds on the maximum achievable throughput, which are tighter than previously known ones. Secondly, we propose the particle-filter-based rate adaptation (PRA), which employs a particle filter to estimate the *a posteriori* channel distribution. The PRA can easily be implemented even when the number of available rates is large. Numerical studies show that the PRA performs within one dB of SNR to the proposed upper bounds for a slowly time-varying channel, even in the presence of multi-user interference.

[1]This work has been accepted for publication in *IEEE Trans. Wireless Commun.*, 2008.

2.1 Introduction

Packet switching is prevalent in current wireless communication systems, e.g., in wireless LANs based on the IEEE 802.11 standards [48] and in cellular networks with 3G long term evolution (LTE) capabilities [33]. Automatic repeat request (ARQ) [6, 7] is commonly used to enhance the reliability or the throughput of packet-switched systems. When the channel experiences an instantaneous deep fade or is subject to strong interference, a packet cannot be recovered. An explicit negative acknowledgement (NACK), or a missing positive ACK (PACK), is then used to signal a retransmission. To efficiently use the channel, the rate at which each packet is encoded, i.e., the modulation constellation and code rate used, should ideally match the instantaneous channel condition. This poses a challenging tracking problem for time-varying channels, particularly in the presence of interference.

Tracking the channel and matching to it the rate of the packet is accomplished by *rate adaptation* [16, 56–58], known also as adaptive signalling [17], adaptive modulation and coding [18], link adaptation [19, 20, 22, 23, 59], auto-rate [21] and adaptive error control [24]. To perform rate adaptation, channel state information (CSI) is needed. Although more informative CSI leads to better channel tracking and hence higher throughput, in practice the availability of CSI is limited by the communication scenario and system employed. For example, in IEEE 802.11a/b/g systems, the Request-to-Send/Clear-to-Send (RTS/CTS) mechanism, a handshaking protocol to set up communication, is exploited to assess the channel state [56] or to differentiate packet collisions from packet failures caused by a deep channel fade [57, 58]. However, the RTS/CTS mechanism is used only in certain communication systems. Some advanced rate adaptation schemes require more extensive feedback beyond the standard ACK feedback, e.g., [17, 18], and may not be compliant even with legacy standards such as IEEE 802.11a/b/g. In frequency division duplex systems, channel reciprocity is often not valid, i.e., the return channel may not behave identically as the forward channel. In such systems, rate adaptation schemes that exploit channel reciprocity for measuring the channel quality of the forward channel, such as [16, 19, 20], cannot be used effectively.

Rate adaptation can be implemented for any ARQ system if only the history of ACKs is used as CSI, such as in [21–23], without assuming channel reciprocity and availability of additional CSI. In [21–23], the rate of the next packet is increased or decreased *relative* to the previous rate, depending on the number of most recent consecutive PACKs or NACKs received. In [24, 25], besides past ACKs, past rates are also used as CSI. No additional feedback is incurred, since the rates are known at the transmitter and need only to be stored in memory. In [24], the CSI is limited to past rates and ACKs in the same *frame*, where a frame typically consists of several packets. In [25], *all* past rates and ACKs are used as CSI, which improves the tracking of the channel quality; for brevity we refer to this as *ACK-rate CSI*.

The problem of optimally adapting the rate using the ACK-rate CSI so as to maximize the throughput is a PSPACE-complete problem [25, 26], which is considered at least as hard as an NP-complete problem. This means that optimal rate adaptation schemes

cannot be computed or implemented practically. Hence, rate adaptation schemes based on heuristics are devised in [25]. However, the complexity of these heuristic schemes can still increase quickly if the number of possible rates becomes large. Since the maximum achievable throughput cannot be computed numerically, a computable upper bound is obtained in [25]. However, this upper bound may not be sufficiently tight in some channels. A tight computable upper bound is desirable since it provides an accurate indication of how close a rate adaptation scheme performs with respect to the optimal one. Moreover, although collisions due to multi-user interference occur frequently in practice, collisions have not been considered in [24, 25]. As such, an overly conservative rate adaptation scheme may result, because an NACK caused by a collision may be wrongly perceived to be caused by a deep channel fade.

In this chapter, to match variations in channel conditions, we employ rate adaptation and seek to maximize the throughput averaged over an infinite time horizon. To limit the feedback, ACK-rate CSI is employed. The ACK feedback is used for rate adaptation in IEEE 802.11a/b/g systems. One of the important challenges in wireless system standardization is to keep the amount of channel feedback small. The use of a one-bit feedback (via ACK) represents the extreme case of limited feedback and is thus useful as benchmark for future schemes or other existing schemes that require more feedback.

To obtain tractable results and to build insights, in our analysis we use a first-order finite-state Markov channel (FSMC) to model the channel variation over time [60]. We assume that the buffer for storing information bits at the transmitter has infinite size and always contains sufficient bits. This is appropriate if many information bits are already pre-stored at the transmitter, such as in streaming applications.

Our contribution pertains to these new improved aspects.

- We study the effects of collisions on rate adaptations, by modeling collisions in the FSMC.
- We establish two new computable upper bounds that are tighter than currently known ones. To obtain these upper bounds, we let the transmitter receive a CSI that is more informative than ACK-rate CSI. Specifically, we periodically update the transmitter with a delayed version of the exact channel coefficient, in addition to the ACK-rate CSI.
- We propose practical near-optimum rate adaptation schemes. To reduce the complexity of real-time implementation, we consider the pragmatic approach of maximizing over a finite time horizon. Further, we propose the *particle-filter-based rate adaptation* (PRA), which employs the particle filter [27] for rate adaptation. The PRA has a complexity that is largely independent of the number of rates used. This allows us to use a large number of rates when we explore the potential of rate adaptation.

For simplicity in obtaining numerical results, we assume that a packet is erroneous if the SNR is less than a rate-dependent threshold or if a collision occurs. Numerical studies show that the throughput performance drops drastically if collisions are not properly accounted for. Moreover, the proposed PRA outperforms conventional

Notation	Meaning
subscript k	time or packet index
$H_k \in \mathcal{S}_H = \{h_1, \cdots, h_N\}$	channel amplitude
$R_k \in \mathcal{S}_R$	rate used for transmission
$\epsilon_k \in \{0, 1\}$	collision (1 if present, 0 if absent)
$A_k \in \{0, 1\}$	ACK (1 if PACK, 0 if NACK)
$\widetilde{H}_k \triangleq \{H, \epsilon\} \in \mathcal{S}_H \times \{0, 1\}$	channel
$C_k \in \mathcal{S}_C$	CSI
$\mathbf{r}_k, \mathbf{a}_k, \widetilde{\mathbf{h}}_k$	$k+1$ by 1 vector with corresponding elements e.g., $\mathbf{a}_k = [A_0, \cdots, A_k]$
 $\bar{\gamma}$ $\bar{\rho}$ q_{ij}	(fixed long term statistics) average SNR power correlation coefficient, collision transition probability from j to i
$t(R_k, \widetilde{H}_k)$	throughput for packet k given channel state $\widetilde{H}_k$
$T(R_k; C_k)$	throughput for packet k given CSI C_k
π, π^*	policy, optimal policy
$\mathcal{T}(\pi), \mathcal{T}^*$	throughput, optimal throughput
$b_k(\widetilde{H}), \mathbf{b}_k \triangleq \{b_k(\widetilde{H}), \forall \widetilde{H}\}$	belief state, collection of all belief states

Table 2.1: Key notations used in this chapter.

schemes and performs within one dB of signal-to-noise ratio (SNR) to the proposed upper bounds for a slowly changing channel, even in the presence of collisions.

Key notations are given in Table 2.1. This chapter is organized as follows. Section 2.2 describes the system model. Section 2.3 formulates the problem of maximizing the throughput averaged over an infinite horizon. To obtain a solution that approaches this throughput, Section 2.4 considers the problem of maximizing the throughput averaged over a sliding window. Section 2.5 then solves this alternative problem using the PRA. Numerical results are presented in Section 2.6. Section 2.7 discusses on the extension of our work. Concluding remarks are given in Section 2.8.

2.2 System Model

The system model is depicted in Fig. 2.1. Over a time horizon of K packets, the CSI available for rate adaptation at time k is denoted as C_k, to be defined in Section 2.2.3. The time index k coincides with the packet index for simplicity. Based on C_k, the rate adaptation block selects a rate $R_k \in \mathcal{S}_R$, in bits per symbol, to transmit packet k. The rate determines the coding rate and modulation scheme used. The buffer collects $R_k N_\mathrm{s}$ information bits from a source which are then encoded as a codeword $\mathbf{x}_k \in \mathbb{C}^{N_\mathrm{s}}$, where N_s is the codeword length. Each codeword uses unit power per symbol on average. The codeword is finally sent as packet k. The bits are encoded and

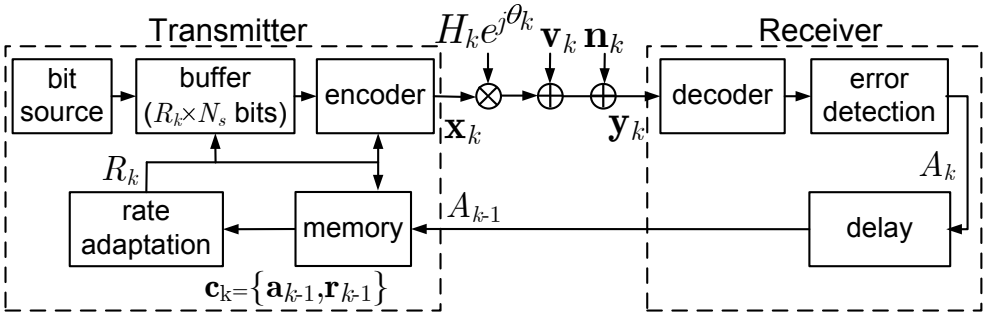


Fig. 2.1: System model for rate adaptation.

decoded independently for each packet, even in retransmissions. This ARQ scheme is commonly known as a Type I hybrid ARQ scheme [8].

We consider a flat-fading channel with (non-negative) channel amplitude $H_k \in \mathbb{R}_+$ that varies (slowly) between packets but is time invariant during each packet duration. The received codeword is

$$\mathbf{y}_k = H_k e^{j\theta_k} \mathbf{x}_k + \mathbf{v}_k + \mathbf{n}_k, \quad k = 1, 2, \cdots, K, \tag{2.1}$$

where $\mathbf{v}_k \in \mathbb{C}^{N_s}$ is multi-user interference and $\mathbf{n}_k \in \mathbb{C}^{N_s}$ is a circularly symmetric complex additive white Gaussian noise (AWGN) vector. The elements in $\mathbf{n}_k$ are independent, each with zero mean and unit variance. We consider coherent detection by a receiver that knows and corrects the channel phase variations, so for simplicity we let $\theta_k = 0$. Without multi-user interference, the average SNR is given by $\bar{\gamma} = \mathbb{E}[H_k^2]$ for all k, assuming that H_k follows a stationary process.

The receiver performs decoding using $\mathbf{y}_k$ with full knowledge of the channel state. Further, the rate is known, say via a packet header. Then, error detection is carried out for the packet, usually by using a cyclic redundancy check (CRC). The receiver sends an ACK bit A_k to the transmitter, either a PACK $A_k = 1$ for correct decoding, or a NACK $A_k = 0$ otherwise. Finally, the transmitter receives A_k with a packet delay, assumed to be received error-free.

To obtain numerical results, we make the following assumptions:

- A1: If multi-user interference is present at time k, packet k is received with error. We say a *collision* has occurred, denoted as $\epsilon_k = 1$.
- A2: If multi-user interference is absent, i.e., $\epsilon_k = 0$, then packet k is received correctly if and only if the rate R_k is below the AWGN channel capacity $\mathsf{C}(H_k) = \log_2\left(1 + H_k^2\right)$.
- A3: Channel amplitudes and collisions occur independently, i.e., H_i is independent of ϵ_j for all i, j.

These assumptions are reasonable if multi-user interference is typically strong (for A1), a capacity-approaching coding scheme is employed (for A2), and all users transmit independently (for A3).

In practice, assumptions A1, A2, A3 may not be fully satisfied, e.g., the frequency of collisions (due to other users' transmissions) may be correlated to one's channel amplitude. Nevertheless, our subsequent results given in Theorems 2.1, 2.2, 2.3 apply generally even if these assumptions are relaxed.

Using assumptions A1, A2, the conditional PACK probability is given by

$$p(A_k = 1|R_k, H_k, \epsilon_k) = \begin{cases} 1, & \epsilon_k = 0 \text{ and } R_k \leq \mathsf{C}(H_k); \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

Here and subsequently, p denotes a probability mass function (pmf). The transmitter may thus infer from a PACK that no collision has occurred *and* R_k is low enough to support the transmission, or from a NACK that a collision has occurred *or* R_k is too high.

2.2.1 A Preview

To build intuition, let us preview the rate outputs of the particle-filter-based rate adaptation (PRA) to be proposed in Section 2.5. For clarity of presentation, a large, finely quantized set of rates $\mathcal{S}_R$ is available for rate adaptation.

In Fig. 2.2, there is no collision, i.e., the collision probability is zero. We see that generally the rate is adapted upwards if a PACK is received, and downwards if a NACK is received. The actual rate used depends on how much we can infer about the channel, by exploiting the available CSI.

Instead, in Fig. 2.3, collision occurs with probability of 0.3. To achieve high throughput, the PRA now takes collision into account and behaves differently compared to the case of no collision. For example, the rate may not necessarily be adapted downwards if a NACK is received. In packets $250 - 260$, a series of NACKs, even for packets transmitted at very low rates, suggests strongly that NACKs are caused primarily by collisions (which turns out to be partially true). Thus, it may be worthwhile to increase the rate, which is done for packets 255 and 258, so that in the event that no collision is present, a high throughput can be achieved.

Intuitively, based on the CSI available, a good rate adaptation algorithm should therefore try to characterize as much as possible the channel by (i) differentiating a collision from a channel fade and (ii) determining the degree of the channel fade.

2.2.2 Channel Statistics

Except for the AWGN, the channel is completely described by the channel amplitude H_k and the collision event ϵ_k. Henceforth, for convenience we formally define the *channel* at time k as $\widetilde{H}_k = \{H_k, \epsilon_k\}$, and the value that the channel takes as the *channel state*. For analytical tractability, we consider a first-order Markovian channel with distribution

$$p(\widetilde{H}_k|\widetilde{H}_0, \cdots \widetilde{H}_{k-1}) = p(\widetilde{H}_k|\widetilde{H}_{k-1}) = p(H_k|H_{k-1})p(\epsilon_k|\epsilon_{k-1}), \tag{2.3}$$

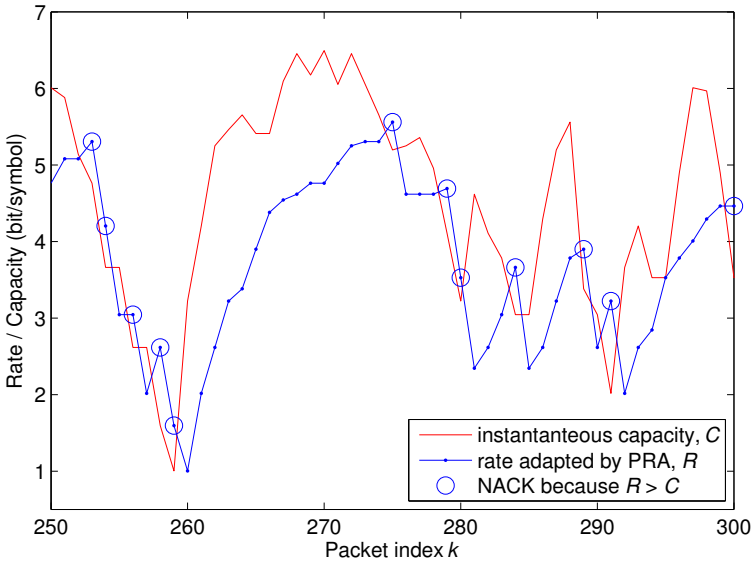


Fig. 2.2: Typical run of the rates adapted using PRA with ACK-rate CSI, zero probability of collision. Parameters: $\bar{\rho} = 0.95$, $\bar{\gamma} = 20$ dB, $q_{10} = 0$, $q_{01} = 1$.

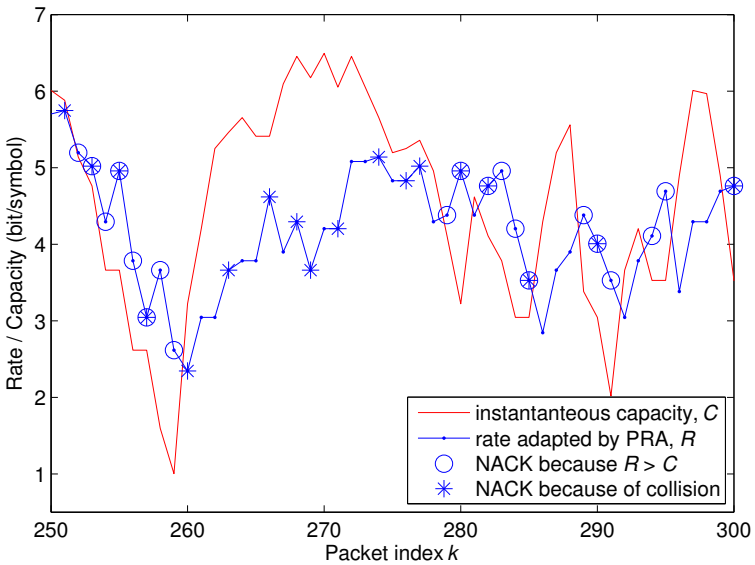


Fig. 2.3: Typical run of the rates adapted using PRA with ACK-rate CSI, 0.3 probability of collision. The same channel amplitudes as in Fig. 2.2 are used. Note that the transmitter cannot differentiate between the causes of the NACKs. Parameters: $\bar{\rho} = 0.95$, $\bar{\gamma} = 20$ dB, $q_{10} = 0.4$, $q_{01} = 0.9$.

for $H_k \in \mathcal{S}_H, \epsilon_k \in \{0,1\}$, where the second equality follows from assumption A3. The subsequent analysis can be straightforwardly extended to an nth-order Markovian channel for $n \geq 2$, by defining the channel instead as

$$\widetilde{H}_k = \{H_k, \cdots, H_{k-n+1}, \epsilon_k, \cdots, \epsilon_{k-n+1}\}.$$

Although an nth-order Markovian channel with larger n better approximates more realistic wireless channels, obtaining numerical results would incur significantly higher complexity (the size of the channel state space increases exponentially with n). We now give details on the statistics of H_k and ϵ_k for the FSMC characterized by (2.3).

2.2.2.1 Channel Amplitude

For our numerical results, we use the FSMC [60] to model temporal variations of H_k. We assume that H_k is in a discrete set $\mathcal{S}_H = \{h_1, \cdots, h_N\}$ with N elements. First, we model the steady-state distribution $p(H_k)$ to be close to $f(G)$, where $f(G)$ is the probability density function (pdf) of the Rayleigh distribution, such that the approximation improves as N increases. To this end, we divide G's support $(0, \infty)$ into N contiguous, non-overlapping parts. Let the nth part be bounded by (τ_{n-1}, τ_n), where $\tau_0 = 0$ and $\tau_N \to \infty$. We choose $\{\tau_n\}$ such that the random variable G is in (τ_{n-1}, τ_n) with the same probability for all n, i.e., $\int_{\tau_{n-1}}^{\tau_n} f(G)\,\mathrm{d}G = 1/N$ for all n. In [60], the nth state h_n is assigned as the mid-point of τ_{n-1} and τ_n, i.e., $h_n = (\tau_{n-1}+\tau_n)/2$. Instead, we assign $h_n = \tau_{n-1}$, which ensures that a PACK occurs only if $R_k \leq \mathsf{C}(H_k = h_n)$. Next, we model the channel-amplitude transition probability $p(H_k|H_{k-1})$ such that $p(H_k = h_j|H_{k-1} = h_i) \propto \int_{\tau_{j-1}}^{\tau_j}\int_{\tau_{i-1}}^{\tau_i} f(G_k|G_{k-1})\mathrm{d}G_{k-1}\mathrm{d}G_k$ (with appropriate normalization so that the pmf sums to one), where the bivariate Rayleigh distribution $f(G_k, G_{k-1})$ is fully determined by the power correlation coefficient [61, Eqn (1)]

$$\bar{\rho} = \mathrm{cov}(G_k^2, G_{k-1}^2)/\sqrt{\mathrm{var}(G_k^2)\mathrm{var}(G_{k-1}^2)}.$$

Further details are found in [60]. The degree of the channel variations is reflected in $\bar{\rho}$: the closer it is to one, the slower the channel variation is. As an example, in Fig. 2.2 we set $\bar{\rho} = 0.95$, where the channel capacity varies as a result of channel fading. We see that the capacity becomes almost uncorrelated after a lag of more than around ten packets.

2.2.2.2 Collision

The collision transition probability is denoted as $q_{ij} \triangleq p(\epsilon_k = i|\epsilon_{k-1} = j)$. Since $\sum_i q_{ij} = 1$ and ϵ_k takes two possible values, the collision statistics is completely specified by q_{10} and q_{01}. The steady-state collision probability can then be obtained as $p(\epsilon = 1) = q_{10}/(q_{01} + q_{10})$. For example, in Fig. 2.3 we (arbitrarily) choose $q_{01} = 0.4, q_{01} = 0.9$, so $p(\epsilon = 1) \approx 0.3$. This relatively high collision probability reflects a challenging scenario: it is ambiguous if an NACK occurs due to a channel fade or to a collision.

In our analysis, for simplicity we assume the parameters that describe the long-term channel statistics, namely $\bar{\gamma}, \bar{\rho}, q_{10}$ and q_{01}, to be known to the transmitter. In practice, these parameters may be tuned based on priori knowledge of the network or estimated online over a long time scale, see e.g. [62].

2.2.3 CSI

We initialize the ACK and rate as $A_0 = \emptyset, R_0 = \emptyset$, respectively, where $\emptyset$ is the null value. The initial channel state $\widetilde{H}_0 = \{H_0, \epsilon_0\}$ is randomly generated based on the steady-state distribution. We collect all ACKs until time k as vector $\mathbf{a}_k \triangleq [A_0, A_1, \cdots, A_k]$, and similarly all rates, channel amplitudes and channel until time k as $\mathbf{r}_k, \mathbf{h}_k, \tilde{\mathbf{h}}_k$, respectively.

We study the maximum achievable throughput when the following CSI $C_k \in \mathcal{S}_C$ is available at the transmitter, while the receiver has full knowledge of the channel state for decoding. The CSI state space $\mathcal{S}_C$ will be clear from the context. Define $C_0 = \emptyset$.

- *ACK-rate CSI*: $C_k = \{A_{k-1}, R_{k-1}, C_{k-1}\}$, or equivalently $C_k = \{\mathbf{a}_{k-1}, \mathbf{r}_{k-1}\}$. This CSI as depicted in Fig. 2.1 is the primary focus of our study. In words, the ACK-rate CSI consists of the most recent rate and ACK and also the past CSI, all available in a causal manner.
- *Full CSI*: $C_k = \widetilde{H}_k$. The instantaneous channel $\widetilde{H}_k$ is provided as the CSI[2].
- *Delayed CSI*: $C_k = \widetilde{H}_{k-1}$. Due to causality, full CSI cannot be provided in practice. Here, a delayed version of the channel (where the delay is one packet long) is provided as the CSI[3].
- *Periodic CSI*: In addition to the ACK-rate CSI, the transmitter is updated periodically with the delayed channel $\widetilde{H}_{k-1}$, with period P. That is,

$$C_k = \begin{cases} \{\widetilde{H}_{k-1}, A_{k-1}, R_{k-1}, C_{k-1}\}, & k \in \mathcal{S}_P, \\ \{A_{k-1}, R_{k-1}, C_{k-1}\}, & k \in \mathcal{S}_P^c, \end{cases} \tag{2.4}$$

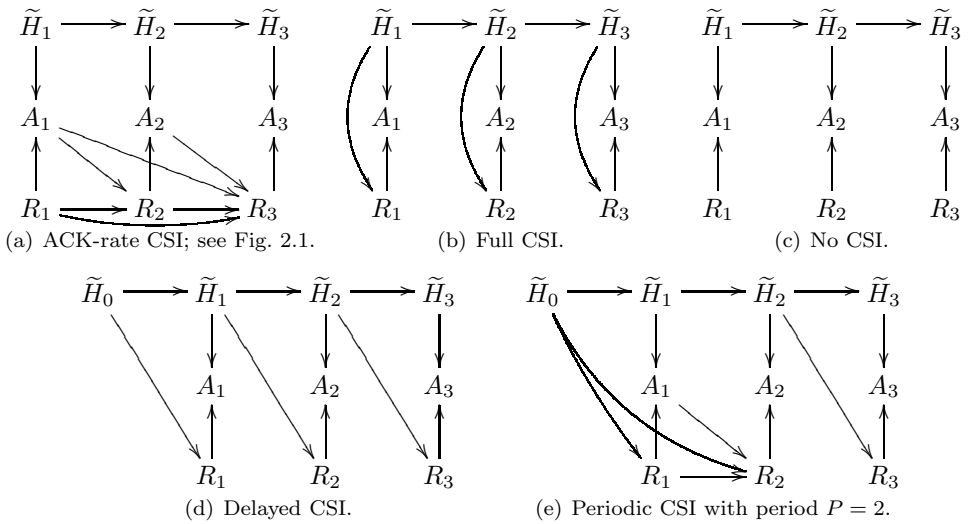
  where $\mathcal{S}_P = \{1, P+1, 2P+1, \cdots\}$ and $\mathcal{S}_P^c$ is its complementary set for positive indices.
- *No CSI*: $C_k = \emptyset$. No CSI is available (besides knowing the channel statistics).

We denote the maximum achievable throughput (to be precisely defined in Section 2.3) corresponding to the above CSI as $\mathcal{T}^*_{\mathrm{ACK-rate}}$, $\mathcal{T}^*_{\mathrm{full}}$, $\mathcal{T}^*_{\mathrm{delayed}}$, $\mathcal{T}^*_{\mathrm{periodic}}$ and $\mathcal{T}^*_{\mathrm{no}}$, respectively. We expect that with more extensive and more informative CSI, a larger maximum throughput can be achieved. Indeed, we will show in Section 2.3 that the maximum achievable throughputs can be ordered accordingly.

To describe the periodic CSI (2.4), it is sufficient to use a past channel $\widetilde{H}_\tau$ and the truncated history of past ACKs and past rates from index τ onwards. Specifically,

[2]The past CSI is not used, which would have given $C_k = \{H_k, C_{k-1}\}$, as no additional information on H_k is obtained.

[3]The past CSI is not used, which would have given $C_k = \{H_{k-1}, C_{k-1}\}$, as no additional information on H_{k-1} is obtained.



(a) ACK-rate CSI; see Fig. 2.1. (b) Full CSI. (c) No CSI.

(d) Delayed CSI. (e) Periodic CSI with period $P = 2$.

Fig. 2.4: Causal diagrams illustrating the dependence of channel $\widetilde{H}_k$, ACK A_k and rate R_k as time progresses during rate adaptation. Different CSIs are available at the transmitter (a)-(e). In all cases, the channel is Markovian and the ACK depends on the rate and channel, while the rate depends on the CSI.

without loss of optimality in achieving $\mathcal{T}^*_{\text{periodic}}$, the periodic CSI (2.4) can be reduced to (see Lemma 2.2 in Section 2.3)

$$\widehat{C}_k = \left\{ \widetilde{H}_{\tau(k)}, [A_{\tau(k)+1}, \cdots, A_{k-1}], [R_{\tau(k)+1}, \cdots, R_{k-1}] \right\}. \tag{2.5}$$

Here, $\tau(k) = P\lfloor (k-1)/P \rfloor$, where $\lfloor x \rfloor$ denotes the largest integer less than x. We interpret $\tau(k)$ as the most recent time index prior to k at which the channel is known exactly. Note that $\widehat{C}_k = \widetilde{H}_{k-1}$ for $k \in \mathcal{S}_P$. To simplify analysis and implementation, we use the reduced form (2.5) over (2.4), unless otherwise stated.

2.2.4 Joint Distribution of Channels, Rates and ACKs

We treat the rate as a random variable, and the sequence of rates as a stochastic process over a time horizon of K packets. The causal relationship of the channel $\{\widetilde{H}_k\}$, the ACKs $\{A_k\}$ and the rates $\{R_k\}$ can be represented with a directed graph [63], which can be established rigourously as a *causal diagram* [64]. Let $\mathcal{A}_k, \mathcal{B}_k$ be sets of random variables at time k or earlier, and let x_k be a random variable at time k. If $p(x_k|\mathcal{A}_k, \mathcal{B}_k) = p(x_k|\mathcal{A}_k)$, we draw an arrow from each of the random variables in $\mathcal{A}_k$ to x_k. Intuitively we may say that x_k is *caused* by the random variables in $\mathcal{A}_k$, but not in $\mathcal{B}_k$.

Besides providing a graphical overview of how the random variables interact, a causal diagram allows any conditional independence to be easily established [63, 64]. The

causal diagrams for different CSIs are illustrated in Fig. 2.4 with $K = 3$, based on the following considerations. For all k, we have $\widetilde{H}_k \to \widetilde{H}_{k+1}$ since the channel is Markovian. Moreover, the probability of a PACK or NACK depends only on the present channel and rate, thus we have $\{\widetilde{H}_k, R_k\} \to A_k$. Finally, we let each rate depend only on its corresponding CSI, so that $C_k \to R_k$. For any CSI, the joint pmf of $\tilde{\mathbf{h}}_K, \mathbf{a}_K, \mathbf{r}_K$ can then be factored as

$$p(\tilde{\mathbf{h}}_K, \mathbf{a}_K, \mathbf{r}_K) = \prod_{k=1}^{K} p(\widetilde{H}_k|\widetilde{H}_{k-1}) p(R_k|C_k) p(A_k|R_k, \widetilde{H}_k). \tag{2.6}$$

2.2.5 Rate Adaptation

Rate adaptation is performed through a *rate-adaptation policy* π, defined by the set of $p(R_k|C_k)$ for all k and all C_k. In practice a (deterministic) function f_k selects the rate to be used at time k, i.e., $R_k = f_k(C_k)$. In this case, the policy is *deterministic*. Then in (2.6), we can substitute for $p(R_k|C_k)$ the Kronecker delta function $\delta(R_k - f_k(C_k))$. Moreover, if $f_k(C_k)$ is independent of k for all C_k, we say that the policy π is *stationary*. Strictly speaking, many types of CSI, like the ACK-rate CSI, do not admit a stationary policy. This is because the size of the CSI grows as time progresses, which necessitates a different f_k (with input of different length) for different k. However, we will show that each CSI can be equivalently mapped into the belief state, which has the same dimension for any k. Thus, a policy for different types of CSI when defined over the belief states may still be stationary.

2.2.6 Throughput

If packet k is received correctly, its contributes an *instantaneous throughput* given by the data rate R_k. This occurs when $A_k = 1$. If $A_k = 0$, the packet is lost in an outage and it is discarded (a common practice in delay-sensitive applications) or retransmitted. In both cases the instantaneous throughput is zero. Given the channel state $\widetilde{H}_k$, the expected throughput for packet k encoded at rate R_k is thus

$$t(R_k, \widetilde{H}_k) = R_k p(A_k = 1|R_k, \widetilde{H}_k). \tag{2.7}$$

If the channel is not known exactly, the expected throughput for packet k given CSI C_k is then

$$T(R_k; C_k) = \mathbb{E}_{\widetilde{H}_k|C_k}[t(R_k; C_k)] = \sum_{\widetilde{H}_k} p(\widetilde{H}_k|C_k)\ t(R_k, \widetilde{H}_k). \tag{2.8}$$

Here, the expectation is performed over the *a posteriori* channel pmf $p(\widetilde{H}_k|C_k)$, which we denote as

$$b_k(\widetilde{H}_k) = p(\widetilde{H}_k|C_k) \tag{2.9}$$

for a given CSI C_k and call this the *belief state*. Given C_k, the set of all belief states $\mathbf{b}_k \triangleq \{b_k(\widetilde{H}) \ \forall \widetilde{H}\}$ is sufficient to compute the expected throughput. Using (2.2), (2.7) and assumption A3, (2.8) becomes

$$T(R_k; C_k) = R_k \Pr\left(R_k < \mathsf{C}(H_k)|C_k\right) p(\epsilon_k = 0|C_k). \tag{2.10}$$

2.3 Maximizing Infinite-Horizon Throughput

In this section, we consider the maximization of the throughput over an infinite time horizon. We derive expressions for the maximum achievable throughput for different types of CSI, and a sequence of inequalities that relate them. In addition, we derive two upper bounds on the throughput achieved with ACK-rate CSI, both of which are tighter than previously known bounds.

2.3.1 Problem Formulation

For any type of CSI, the *long-term throughput* $\mathcal{T}(\pi)$ given policy π is obtained by averaging the expected throughput over an infinite-time horizon, i.e.,

$$\mathcal{T}(\pi) = \lim_{K\to\infty} \frac{1}{K} \mathbb{E}\left[\sum_{k=1}^{K} T(R_k; C_k)\right]. \tag{2.11}$$

The expectation is defined with respect to the joint pmf (2.6). In our study where H is always bounded, $\mathcal{T}(\pi)$ is likewise always bounded. We take the limit $\lim_{K\to\infty}$ to be $\liminf_{K\to\infty}$ if the limit does not exist.

Using the ACK-rate CSI $C_k = [\mathbf{a}_{k-1}, \mathbf{r}_{k-1}]$, the *maximum achievable throughput* obtained by the optimal rate adaptation policy π^* is denoted by

$$\mathcal{T}^*_{\mathrm{ACK-rate}} = \max_{\pi} \mathcal{T}(\pi) = \mathcal{T}(\pi^*). \tag{2.12}$$

In general, the superscript * denotes optimality while the subscript denotes the type of CSI used.

Our objective is to find a rate adaptation policy using the ACK-rate CSI that achieves a throughput close to $\mathcal{T}^*_{\mathrm{ACK-rate}}$. In addition we wish to obtain tight upper bounds for $\mathcal{T}^*_{\mathrm{ACK-rate}}$ that can be computed.

2.3.2 Main Analytical Results and Discussions

The maximum achievable throughput, given by (2.12) for ACK-rate CSI, is also defined similarly for other types of CSIs by an appropriate substitution of C_k. In

Section 2.3.3, we obtain analytical expressions for the maximum achievable throughput for full CSI, delayed CSI, periodic CSI and no CSI, denoted respectively as $\mathcal{T}^*_{\text{full}}, \mathcal{T}^*_{\text{delayed}}, \mathcal{T}^*_{\text{periodic}}$ and $\mathcal{T}^*_{\text{no}}$, as summarized in Theorem 2.1.

We say a rate-adaptation policy is *myopic* if for every packet, the rate is adapted to maximize only the current expected throughput, without concerns about the effect on future achievable throughput. The throughput achieved by a myopic policy with CSI C_k is thus $\mathbb{E}_{C_k}\left[\max_{R_k} T(R_k; C_k)\right]$.

Theorem 2.1. *The maximum achievable throughput for full CSI, delayed CSI or no CSI is achieved by a stationary myopic policy, which can be expressed respectively as*

$$\begin{aligned}
\mathcal{T}^*_{\text{full}} &= \mathbb{E}_{\widetilde{H}_k}\left[\max_{R_k} T(R_k; \widetilde{H}_k)\right] && \text{(2.13a)}\\
&\overset{\textit{assume}\ \mathsf{A3}}{=} q_0 \mathbb{E}_H[\mathsf{C}(H)],\\
\mathcal{T}^*_{\text{delayed}} &= \mathbb{E}_{\widetilde{H}_{k-1}}\left[\max_{R_k} T(R_k; \widetilde{H}_{k-1})\right] && \text{(2.13b)}\\
&\overset{\textit{assume}\ \mathsf{A3}}{=} q_0 \mathbb{E}_{H_{k-1}}\left[\max_{R_k} R_k \Pr\left(R_k < \mathsf{C}(H_k) | H_{k-1}\right)\right],\\
\mathcal{T}^*_{\text{no}} &= \mathbb{E}_{\emptyset}\left[\max_{R_k} T(R_k; \emptyset)\right] && \text{(2.13c)}\\
&\overset{\textit{assume}\ \mathsf{A3}}{=} q_0 \max_{R} R_k \Pr\left(R_k < \mathsf{C}(H_k)\right),
\end{aligned}$$

where we denote $q_0 = p(\epsilon = 0)$. The maximum achievable throughput for periodic CSI with period P can be expressed as

$$\mathcal{T}^*_{\text{periodic}} = \mathbb{E}_{C_1}\left[J_1(C_1)\right]/P, \tag{2.13d}$$

where J_1 can be expressed recursively with decreasing $k = P, P-1, \cdots, 1$ according to

$$J_k(C_k) = \max_{R_k} T(R_k; C_k), \qquad k = P \tag{2.14a}$$

$$J_k(C_k) = \max_{R_k} \left\{T(R_k; C_k) + \mathbb{E}_{C_{k+1}|C_k}\left[J_{k+1}(C_{k+1})\right]\right\}, \; k = P-1, \cdots, 1. \tag{2.14b}$$

Here in (2.13) and (2.14), the maximization of the rate R_k is carried out over the discrete set $\mathcal{S}_R$.

Proof. We employ Bellman's equations [65] for our proof; see Section 2.3.3 for details. □

We remark that the expressions (2.13) for full CSI, delayed CSI and delayed CSI are straightforward to compute, given the channel distribution. The expressions (2.14) for periodic CSI can be computed for small P, but the complexity grows exponentially with the period P.

In [25], $\mathcal{T}^*_{\text{delayed}}$ has been used as an upper bound for $\mathcal{T}^*_{\text{ACK-rate}}$. Our second main result is Theorem 2.2, which introduces an upper bound $\mathcal{T}^*_{\text{periodic}}$ that is tighter than

$\mathcal{T}^*_{\text{delayed}}$. Theorem 2.3 introduces another upper bound $\mathcal{T}_{\text{ub}}$ that is also tighter than $\mathcal{T}^*_{\text{delayed}}$. The superscript * is omitted in this notation $\mathcal{T}_{\text{ub}}$, because the policy that achieves $\mathcal{T}_{\text{ub}}$ is genie-aided and cannot be implemented in practice. From numerical simulations in Section 2.6, $\mathcal{T}_{\text{ub}}$ can be even tighter than $\mathcal{T}^*_{\text{periodic}}$.

Theorem 2.2. *The maximum achievable throughput for full CSI, delayed CSI, periodic CSI, ACK-rate CSI and no CSI are ordered decreasingly, i.e.,*

$$\mathcal{T}^*_{\text{full}} \geq \mathcal{T}^*_{\text{delayed}} \geq \mathcal{T}^*_{\text{periodic}} \geq \mathcal{T}^*_{\text{ACK-rate}} \geq \mathcal{T}^*_{\text{no}}. \tag{2.15}$$

Proof. We rely on (2.6) implicitly and on Theorem 2.1; see Appendix 2.B for details. □

We now obtain an alternative upper bound for $\mathcal{T}^*_{\text{ACK-rate}}$. First, let $\mathcal{T}_{\text{ub}}$ be defined by

$$\mathcal{T}_{\text{ub}} = \mathbb{E}_{\widetilde{H}_{k-2}}\left[\max_{R_{k-1}} \mathbb{E}_{R_{k-1},A_{k-1}|\widetilde{H}_{k-2}}\left[\max_{R_k} T(R_k;\bar{C}_k)\right]\right], \tag{2.16}$$

where $\bar{C}_k \triangleq \{R_{k-1}, A_{k-1}, \widetilde{H}_{k-2}\}$. We interpret (2.16) as a maximization of the throughput $T(R_k;\bar{C}_k)$ at time k with CSI $\bar{C}_k$. This CSI consists of the past rate, past ACK and a channel amplitude delayed by *two* units of time. We note that the past rate R_{k-1} *had* been optimized given CSI $\widetilde{H}_{k-2}$ which is relatively delayed by *one* unit of time.

Theorem 2.3. *$\mathcal{T}_{\text{ub}}$ is an upper bound for the maximum achievable throughput with ACK-rate CSI $\mathcal{T}^*_{\text{ACK-rate}}$. Moreover, it is a tighter upper bound than $\mathcal{T}^*_{\text{delayed}}$. That is,*

$$\mathcal{T}^*_{\text{delayed}} \geq \mathcal{T}_{\text{ub}} \geq \mathcal{T}^*_{\text{ACK-rate}}. \tag{2.17}$$

Proof. See Appendix 2.C for a proof. □

Intuitively, two aspects make $\mathcal{T}_{\text{ub}}$ achieve a higher throughput than $\mathcal{T}^*_{\text{ACK-rate}}$. Firstly, the CSI $\bar{C}_k$ available for adapting R_k is more informative than in the case of ACK-Rate CSI, with $\widetilde{H}_{k-2}$ being the additional CSI. Secondly, both past and current rates are used to maximize the current throughput (for packet k), without regarding how past throughput (for packet $k-1$) and future throughput (for packet $k+1$ onwards) are affected. Since past and present rates are *always* used to optimize for the current packet, this policy is genie-aided and cannot be implemented in practice.

2.3.3 Maximum Achievable Throughput

We now obtain expressions for the maximum achievable throughput for various CSI types in Theorem 2.1. We refer to the value that a CSI takes as a CSI state. We now state an important result from optimal control that is useful for subsequent derivations.

Lemma 2.1. *Suppose that for all initial CSI states and all policies, there exists at least one CSI state in its state space $\mathcal{S}_C$ that is visited at least once with positive probability within some bounded time. Then, the maximum achievable throughput $\mathcal{T}^*$ obtained by maximizing (2.11) satisfies Bellman's equation*

$$\mathcal{T}^* + h(C) = \max_R \left\{ T(R;C) + \sum_{C' \in \mathcal{S}_C} p(C'|C,R) h(C') \right\} \tag{2.18}$$

for all CSI states C in $\mathcal{S}_C$. Here, $h(C)$ is an auxiliary function known as the differential reward function[4] *and $p(C|C',R)$ is the transition probability from state C to state C' given rate R. Moreover, a rate adaptation policy where the rate R satisfies (2.18) for all CSI states C in $\mathcal{S}_C$ is optimal, i.e., this stationary policy achieves $\mathcal{T}^*$.*

Proof. See Section 7.4 of [65], rephrased for our throughput maximization problem. □

To prove Theorem 2.1, we assume that $q_{01} > 0$ and $q_{10} > 0$, hence collision occurs with non-zero probability. The case of $q_{01} = q_{10} = 0$ where collision never occur[5] can be proved similarly, if we let $\epsilon = 0$ and define the channel to consist only of the channel amplitude, i.e., $\widetilde{H} = \{H\}$.

2.3.3.1 Full CSI

For full CSI, the CSI state space is $\mathcal{S}_C = \mathcal{S}_H \times \{0,1\}$, where $\mathcal{S}_H$ is the channel state space and $\{0,1\}$ is the collision state sapce. Since any CSI state transits to another CSI state with a positive probability for any policy, Lemma 2.1 applies. By letting $C = \widetilde{H}_k$ and $C' = \widetilde{H}_{k+1}$, we have $p(C'|C,R) = p(C'|C)$ because $\widetilde{H}_k$ is Markovian, thus Bellman's equation becomes

$$\mathcal{T}^* + h(C) = \max_R \left\{ T(R;C) \right\} + \sum_{C' \in \mathcal{S}_C} p(C'|C) h(C'). \tag{2.19}$$

The above maximization needs only to be performed for the function T, independent of the differential reward function h. This shows that a (stationary) myopic policy achieves the maximum throughput in (2.19) and is thus optimal according to Lemma 2.1. Hence, the maximum achievable throughput is given by (2.13a) for full CSI. The second inequality in (2.13a) is obtained using (2.10).

From the above derivations, it follows that the myopic policy is optimal for any CSI type if $p(C'|C,R_k) = p(C'|C)$ for all C. This means that the rate R_k will not affect the future CSI state C' nor the future throughput (which depends on C'), thus intuitively we can focus on maximizing only the current throughput.

[4] Typically the differential reward function has to be solved jointly with $\mathcal{T}^*$ using Bellman's equation, so as to obtain $\mathcal{T}^*$.

[5] The case when collision occurs with probability one is clearly not interesting.

2.3.3.2 Delayed CSI

For delayed CSI, the CSI state space is the same as for full CSI, so Lemma 2.1 applies. In this case, we let $C = H_{k-1}$ and $C' = H_k$. Similarly $p(C'|C, R_k) = p(C'|C)$ for all C, so the myopic policy is again optimal. Hence, we obtain (2.13b), where the second inequality is obtained using (2.10).

2.3.3.3 No CSI

For no CSI, the CSI state space consists of only the null value $\emptyset$. Hence, we have $p(C'|C, R_k) = p(C'|C)$ trivially and so the myopic policy is optimal. Thus, we use a fixed rate to maximize $\mathbb{E}_H[T(R;\emptyset)]$ for all packets. Hence, we obtain (2.13c), where the second inequality is obtained using (2.10).

2.3.3.4 Periodic CSI

For periodic CSI (and also ACK-rate CSI), myopic optimization is generally sub-optimal. The following lemma, however, allows us to simplify analysis by using the reduced periodic CSI (2.5), instead of the original periodic CSI (2.4).

Lemma 2.2. *The maximum achievable throughput for periodic CSI (2.4) is the same as the maximum achievable throughput for reduced CSI (2.5).*

Proof. For clarity, let us denote, at time k, the periodic CSI as $\widetilde{C}_k$ and the reduced CSI as $\widehat{C}_k$. To show that the maximum achievable throughput is the same for both CSIs, it is sufficient to show that the belief state $b_k = p(\widetilde{H}_k|C_k)$ is the same for both CSIs $C_k = \widetilde{C}_k$ and $C_k = \widehat{C}_k$ for all k. This is because the belief state serves as a sufficient input for a policy to determine the next rate (whether the policy is optimal or not).

Using the Markov property that $\widetilde{H}_k \to \widetilde{H}_{k+1}$ and $\{\widetilde{H}_k, R_k\} \to A_k$, we obtain

$$p(\widetilde{H}_k|\widetilde{C}_k) = \begin{cases} p(\widetilde{H}_k|\widetilde{H}_{k-1}), & k \in \mathcal{S}_P; \\ \sum_{\widetilde{H}_{k-1}} p(\widetilde{H}_{k-1}|\widehat{C}_k) p(\widetilde{H}_k|\widetilde{H}_{k-1}), & k \in \mathcal{S}_P^c, \end{cases} \tag{2.20}$$

where $\widehat{C}_k$ is given by (2.5) for $k \in \mathcal{S}_P^c$. Moreover, it can be verified that $p(\widetilde{H}_k|\widehat{C}_k)$ is also given by the right-hand side of (2.20). Since $p(\widetilde{H}_k|\widetilde{C}_k) = p(\widetilde{H}_k|\widehat{C}_k)$, it follows that the maximum achievable throughput is the same for both periodic and reduced CSIs. □

Lemma 2.2 shows that past CSI prior to $\tau(k)$ can be discarded, but after $\tau(k)$, when the channel amplitude is not yet exactly known, the subsequent CSI still needs to be retained. This result is intuitively reasonable: since the channel is modeled as Markovian, the knowledge of an exact channel amplitude makes priori CSI redundant. We can thus consider the rate adaptation policy over a single period: the first packet

in the period receives a CSI consisting of an independent realization of the delayed channel amplitude, while the remaining $P-1$ packets receive an ACK-rate CSI. The throughput over an infinite-time horizon is then equal to the throughput averaged over this period, i.e.,

$$T_{\text{periodic}}(\pi) = \frac{1}{P}\mathbb{E}\left[\sum_{k=1}^{P} T(R_k; \widehat{C}_k)\right], \tag{2.21}$$

where $\widehat{C}_k$ is given by (2.5). Here, we assume that a periodic policy is used. That is, the rate is adapted in the same way for all packets spaced apart by period P. An optimal policy, if it exists, is also given by a periodic policy. This is because if we have a non-periodic optimal policy, we can choose to repeat the policy over the particular period that maximizes the throughput (2.21); this new periodic policy gives the same throughput, or higher. This justifies us to focus on one period of a periodic policy.

Let J_k be the maximum throughput accumulated over packet k to packet P, given CSI state C_k at time k. The maximum achievable throughput (2.21) can then be obtained from Bellman's equation *with finite time horizon* [65], given by

$$\mathcal{T}^*_{\text{periodic}} = \max_{\pi} T_{\text{periodic}}(\pi) = \mathbb{E}_{C_1}\left[J_1(C_1)\right]/P \tag{2.22}$$

where J_k is given by (2.14). To obtain $J_1(C_1)$ and hence $\mathcal{T}^*_{\text{periodic}}$, we can perform a backward recursion. This is done by first obtaining (2.14a) for all C_P, and then obtaining (2.14b) for all C_k with decreasing $k = P-1, \cdots, 1$. The overall complexity is dominated by the first recursion, whose complexity increases exponentially with P. However, computation is still feasible for small P.

2.4 Maximizing Sliding-Horizon Throughput

To find the optimal policy π^* with ACK-rate CSI is a PSPACE-complete problem even if the horizon is finite [26]. A PSPACE-complete problem is solved using a polynomial amount of memory and unlimited time and is considered at least as hard as an NP-complete problem. To obtain an implementable policy that achieves close to the maximum achievable throughput $\mathcal{T}^*_{\text{ACK}-\text{rate}}$, this section solves an alternative problem by considering a finite time horizon. Although the solution for this alternative problem still cannot be implemented exactly, it allows a highly accurate approximate solution to be realized via a particle filter, which will be considered in Section V.

2.4.1 Problem Formulation

For FSMC, the autocorrelation function of the channel amplitude appears to decrease exponentially with increasing time lag [60]. Hence, the validity of the information provided by the CSI diminishes rapidly into the future. As such, there may be little

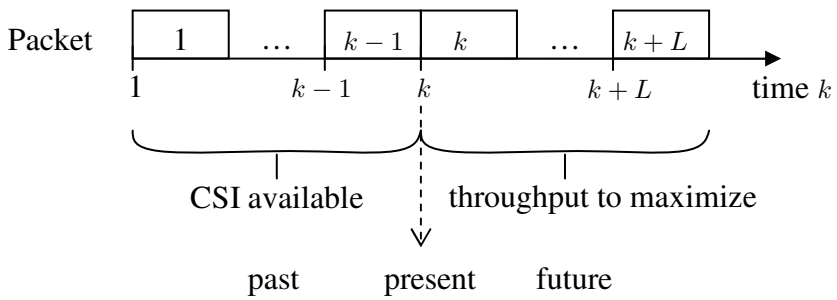


Fig. 2.5: Using CSI from the past to maximize throughput in the future.

loss if a rate adaptation policy maximizes average throughput over a *finite horizon*, even though in the original problem the horizon is infinite.

We thus consider an alternative problem by limiting the horizon, see Fig. 2.5. At time k (the present), we are given the ACK-rate CSI C_k (from the past). We wish to maximize the (future) throughput of next $L+1$ packets given by

$$T_{\mathrm{SH}}(\pi_k; C_k) = \frac{1}{L+1}\mathbb{E}\left[\sum_{l=k}^{k+L} T_l(R_l; C_l)\right] \tag{2.23}$$

by varying the policy π_k consisting of rates of packets $k, \cdots, k+L$. We call T_{SH} the *sliding-horizon throughput* since as time k progresses, the time horizon shifts forward. From the optimum policy π_k, we then use the optimum rate R_k^{o} corresponding to packet k for transmission, i.e.,

$$R_k^{\mathrm{o}} = \arg\max_{R_k}\left\{\max_{\pi_k \backslash R_k} T_{\mathrm{SH}}(\pi_k; C_k)\right\} \tag{2.24}$$

where $\pi_k \backslash R_k$ refers to the rates in policy π_k except for R_k. Thus, we may treat the future rates as auxiliary variables which are tentatively optimized but may be discarded once R_k^{o} is obtained. Next, at time $k+1$, the ACK A_{k+1} is received and the CSI C_{k+1} is updated. The process of obtaining R_k^{o} based on (2.23), (2.24) with k replaced by $k+1$ is then performed, and so on for subsequent packets as the next ACK is received.

In (2.23), (2.23), L is taken as a fixed parameter. As L increases, the effects of rate adaptation on future throughput are better taken into account so the corresponding throughput is expected to improve. When L approaches infinity, R_k maximizes the throughput averaged over an infinite-time horizon and hence achieves the maximum throughput $\mathcal{T}^*$. In the remainder of this chapter, we consider $L = 0, 1$. Extensions for larger L can be carried out similarly but with an exponential increase in the complexity of optimization.

2.4.2 Myopic Optimization: $L = 0$

We defer the implementation details for $L = 1$ to the next section. For $L = 0$, we do not need to consider how future throughput is affected. Thus, given C_k we use a myopic policy according to $R_k = \arg\max_{R_k} \mathbb{E}\left[T(R_k; C_k)\right]$. This policy can be shown to be equivalent to the Q-MDP policy considered in [25], as follows.

The Q-MDP policy is based on a general heuristic strategy first proposed in [66]. This policy solves another alternative problem, in which ACK-rate CSI is available for the present packet but full CSI is assumed to be available in the next immediate packet. That is, $C_k = [\mathbf{a}_{k-1}, \mathbf{r}_{k-1}]$ is given at time k while $C_{k+1} = \widetilde{H}_{k+1}$ (not yet known at time k) *will* be given at time $k+1$. The rate R_k is then chosen to maximize the throughput of packet $k, k+1, \cdots$. We note that R_k cannot affect the future CSI $C_i, i \geq k+1$, given that the channel is known exactly, nor affect the future instantaneous throughput of packet i. Thus, the Q-MDP policy reduces to the myopic policy where only the throughput of packet k is maximized. This conclusion holds in this chapter where the bit buffer is never empty. In [25] where the buffer can be empty, however, the Q-MDP and the myopic policies can be different. This is because the CSI state includes the (limited) buffer queue length, and hence the future CSI and throughput can be affected by R_k.

2.5 Particle-Filter-Based Rate Adaptation (PRA)

This section shows that the implementation of maximizing the sliding-horizon throughput for ACK-rate CSI is mainly determined by how the belief states are stored and maintained over time. To reduce the high implementation complexity, we propose using the particle filter. Before we introduce the particle filter in Section 2.5.2, we first analyze the bottlenecks in directly computing the sliding-window throughput.

2.5.1 Direct Computation

Consider $L = 0$. The optimal rate adaptation policy is obtained by directly maximizing the throughput $T(R_k; C_k)$, which can be computed given the belief states.

Consider $L = 1$. The policy π_k consists of the rate R_k for the current packet and also of *both* rates $R_{k+1}(A_k = 0), R_{k+1}(A_k = 1)$ for the next packet, depending on the yet-to-be-known ACK A_k. We can express (2.23) as

$$\begin{aligned} 2T_{\mathrm{SH}}(\pi_k; C_k) &= T_k(R_k; C_k) + \mathbb{E}_{A_k|C_k,\pi_k}\left[T_{k+1}(R_{k+1}(A_k); C_{k+1})\right] && (2.25)\\ &= T_k(R_k; C_k) + \mathbb{E}_{\widetilde{H}_k, A_k|C_k,\pi_k}\left[T_{k+1}(R_{k+1}(A_k); C_{k+1})\right] && (2.26)\\ &= T_k(R_k; C_k) + \mathbb{E}_{\widetilde{H}_k|C_k}\mathbb{E}_{A_k|\widetilde{H}_k,R_k}[T_{k+1}(R_{k+1}(A_k); C_{k+1})]. && (2.27) \end{aligned}$$

Here, (2.25) follows since A_k is the remaining random variable in (2.23) given C_k, π_k, (2.26) follows from introducing the channel $\widetilde{H}$ which is a hidden random variable and

(2.27) follows from the joint pmf (2.6).

The two terms in (2.27) are the expected throughput of packet $k, k+1$, respectively, given CSI C_k and policy π_k. We write (2.27) in terms of the belief states $\{b_{k+1}(\widetilde{H}_k)\}$ corresponding to the CSI C_k, and the belief states $\{b_{k+1}(\widetilde{H}_{k+1}, A_k, R_k)\}$ corresponding to the future CSI $C_{k+1} = [A_k, R_k, C_k]$, (2.8) can be expressed as

$$2T_{\mathrm{SH}}(\pi_k; C_k) = \sum_{\widetilde{H}_k} b_k \Bigg(t_k(R_k; C_k) + \mathbb{E}_{A_k|\widetilde{H}_k, R_k} \Bigg[\sum_{\widetilde{H}_{k+1}} b_{k+1}(A_k)\, t_{k+1}(R_{k+1}(A_k); C_{k+1}) \Bigg] \Bigg). \quad (2.28)$$

For brevity we subsequently omit the argument $\widetilde{H}_k$ in $b_k(\widetilde{H}_k), b_{k+1}(\widetilde{H}_{k+1}, A_k, R_k)$. Note that t_k can be obtained using (2.7). To compute the throughput (2.28), what remain to obtain are the belief states.

2.5.1.1 Maintaining the Belief States

The Markov property of the channel allows the belief state b_k to be obtained recursively, by using the *prediction* and *update* steps, given as

$$b_k = p(\widetilde{H}_k|C_k) = \sum_{\widetilde{H}_{k-1}} p(\widetilde{H}_k|\widetilde{H}_{k-1}) p(\widetilde{H}_{k-1}|C_k), \quad (2.29)$$

$$\begin{aligned} p(\widetilde{H}_{k-1}|C_k) &\propto p(\widetilde{H}_{k-1}|\mathbf{a}_{k-2}, \mathbf{r}_{k-1})\; p(A_{k-1}|\widetilde{H}_{k-1}, \mathbf{a}_{k-2}, \mathbf{r}_{k-1}) \\ &= b_{k-1}\; p(A_{k-1}|\widetilde{H}_{k-1}, R_{k-1}), \end{aligned} \quad (2.30)$$

respectively. The last line of the update step results from (2.6), or by exploiting the conditional independence shown in Fig. 2.4(a) [67].

Recall that $\mathbf{b}_k$ is the set of the belief states over all $\widetilde{H}_k$. To compute a specific belief state b_k in $\mathbf{b}_k$, clearly from (2.29), (2.30) we need to only maintain in memory $\mathbf{b}_{k-1}, R_{k-1}, A_{k-1}$. Consequently, without any loss of information, we can discard all past beliefs $\mathbf{b}_1, \cdots, \mathbf{b}_{k-2}$ and all past CSIs except for R_{k-1}, A_{k-1}. We note that $\mathbf{b}_{k+1}$, which is used to predict the channel at time $k+1$ in (2.28), can also be computed using $\mathbf{b}_k$ for a given (tentative) rate R_k and ACK A_k.

2.5.1.2 Issues with Maintaining Belief States

There are two advantages of maintaining the most current belief states and ACK-rate CSI, instead of all ACK-rate CSIs. First, the belief states allow direct computation of the throughput. Second, the space of the belief state is fixed, while the space of the CSI grows as time progresses. The disadvantage is that the computation in the prediction step is complex if the number of channel states N is large. Moreover, it is impossible

to accurately keep the belief state in memory, as $\mathbf{b}_k$ lies in a (real) probability space of dimension $N-1$. To solve these problems, we consider an approximate but highly accurate technique based on a sequential Monte Carlo method, known as the particle filter [68].

2.5.2 Proposed Computation via Particle Filter

We employ the particle filter to maintain the belief states and to estimate the throughput. The particle filter is a sequential Monte Carlo method that estimates a pmf or pdf by a recursive importance sampling of random samples, known as particles. Particle filters are popularized by the sampling importance resampling (SIR) filter in [27].

At time k, the particle filter maintains in memory the random measure

$$\chi_k = \left\{ \left(\widetilde{H}_k^{(n)}, w_k^{(n)} \right), n = 1, \cdots, N_p \right\},$$

where $\widetilde{H}_k^{(n)}$ is the nth particle with weight $w_k^{(n)}$. For initialization, we may independently generate $\widetilde{H}_0^{(n)}$ according to the probability $p(\widetilde{H})$ and fix $w_0^{(n)} = 1/N_p$ for all n. The random measure χ_k forms an estimate of the belief states at time k according to

$$b_k \approx \sum_{n=1}^{N_p} w_k^{(n)} \, \delta(\widetilde{H}_k - \widetilde{H}_k^{(n)}). \tag{2.31}$$

The random measure χ_k is generated recursively over time via Monte Carlo sampling [27], based on the knowledge of the system dynamics (governed by the probabilities $p(\widetilde{H}_k|\widetilde{H}_{k-1})$, $p(A_{k-1}|\widetilde{H}_{k-1}, R_{k-1})$), and on the latest observed acknowledgement and rate. Specifically, given $\widetilde{H}_{k-1}^{(n)}$, we independently generate $\widetilde{H}_k^{(n)}$ with the importance sampling function $p(\widetilde{H}_k|\widetilde{H}_{k-1} = \widetilde{H}_{k-1}^{(n)})$. Then, given $w_{k-1}^{(n)}$, we assign the weight of the nth particle as $w_k^{(n)} = w_{k-1}^{(n)} p(A_{k-1}|\widetilde{H}_{k-1} = \widetilde{H}_{k-1}^{(n)}, R_{k-1})$. Finally, normalization is carried out so that all weights in time k sum to unity. From χ_{k-1} we can thus obtain χ_k. Similarly, from χ_k we can obtain χ_{k+1} given (tentative) A_k, R_k. Then, χ_{k+1} is used to approximate the belief state at time $k+1$ as

$$b_{k+1}(A_k, R_k) \approx \sum_{m=1}^{N_p} w_{k+1}^{(m)}(A_k, R_k) \, \delta(\widetilde{H}_{k+1} - \widetilde{H}_{k+1}^{(m)}). \tag{2.32}$$

Finally, substituting (2.7), (2.31) and (2.32) into (2.28) allows the throughput to be approximated as

$$\begin{aligned} 2T_{\mathrm{SH}}(\pi_k; C_k) \approx \sum_{n=1}^{N_p} w_k^{(n)} \Bigg(& p(A_k = 1|R_k, \widetilde{H}_k = \widetilde{H}_k^{(n)}) \\ & + \mathbb{E}_{A_k|\widetilde{H}_k, R_k} \left[\sum_{m=1}^{N_p} w_{k+1}^{(m)}(A_k) \, p(A_k = 1|R_k, \widetilde{H}_k = \widetilde{H}_k^{(m)}) \right] \Bigg). \end{aligned} \tag{2.33}$$

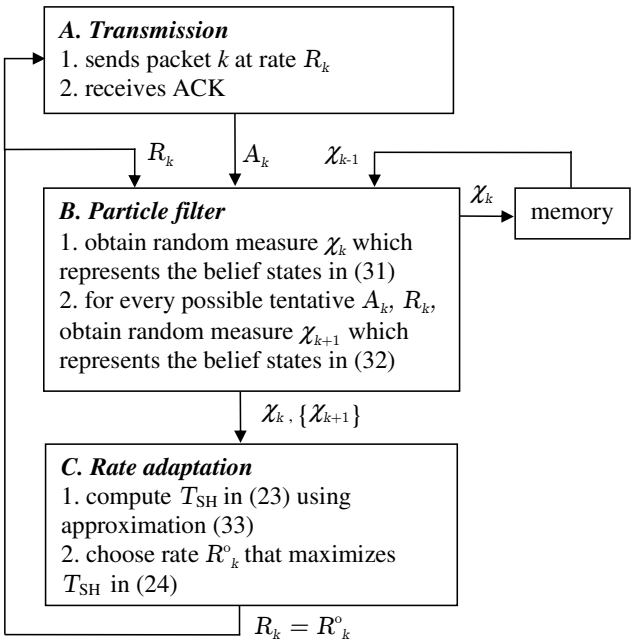


Fig. 2.6: Summary of implementation of PRA for $L = 1$.

This approximation improves as the total number of particles N_p is increased.

Fig. 2.6 highlights the key steps to implement the PRA for rate adaptation. Although we are initially given the ACK-rate CSI $C_k = \{R_k, A_k, C_{k-1}\}$, we see from Fig. 2.6 that $\{R_k, A_k, \chi_k\}$ is used as input to the particle filter. As $N_p \to \infty$, this input becomes sufficient for throughput maximization (as the approximation (2.33) becomes accurate) and may then be treated to be equivalent to the ACK-rate CSI.

Particle filters may experience the degeneracy phenomenon [68], where all but one particle will have negligible weight after several recursions. Solutions to circumvent this problem are described in [68]. We follow [27] by resampling the particles, which effectively normalizes the weights uniformly after every recursion. In our simulations, we did not see the degeneracy phenomenon over runs of 1000 packets.

2.6 Numerical Study

For our numerical studies, we discretize the channel amplitude $H \in \mathcal{S}_H$ by using $N = 100$ channel states, as described in Section 2.2.2. To reduce the effects of rate quantization and to observe the full dynamic behavior of rate adaptation, we match the set of available rates $\mathcal{S}_R$ to the channel state space, according to $\mathcal{S}_R = \{\mathsf{C}(H), H \in \mathcal{S}_H)\}$. To give accurate results we use $N_p = 1000$ particles.

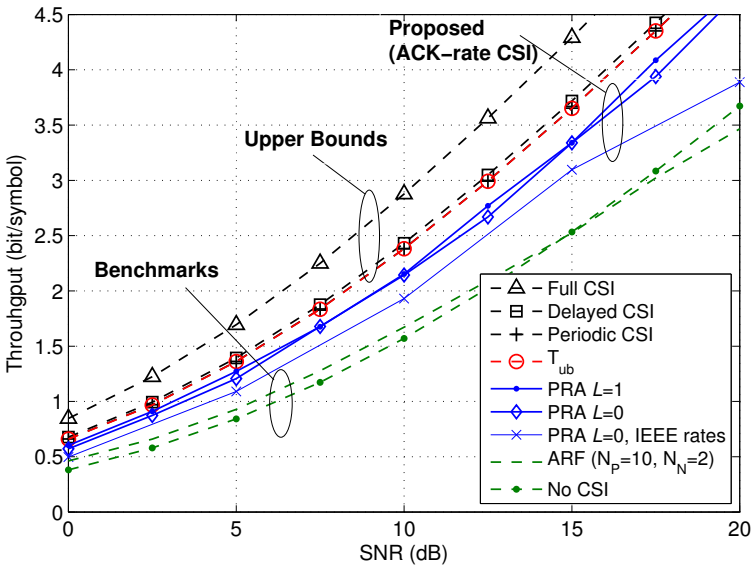


Fig. 2.7: The PRA compared to the benchmarks and upper bound. Parameters: $\bar{\rho} = 0.99$, $q_{10} = 0$, $q_{01} = 1$.

Typical runs of the PRA, without and with colllisions, have been shown in Figs. 2.2 2.3, respectively. The channel amplitudes (same in both figures) are generated randomly at an average SNR of $\bar{\gamma} = 20$ dB and a power correlation coefficient of $\bar{\rho} = 0.95$. In the latter case, NACKs may be due to collisions. As mentioned in Section 2.2, the PRA can account for collision and behave differently if collisions are present.

The long-term throughput for ACK-rate CSI is obtained by Monte Carlo simulations: we average the instantaneous throughput from packet $k = 100$ (after steady state) to packet $k = 1000$, then average again over 200 simulation runs. On the other hand, the maximum achievable throughput for other CSI is calculated analytically according to Section 2.3.3. We vary $\bar{\gamma}$ over the practical range of 0 dB to 20 dB.

We consider the case of either no collision ($q_{10} = 0$, $q_{01} = 1$) or collision ($q_{10} = 0.4$, $q_{01} = 0.9$), and either moderate fading ($\bar{\rho} = 0.95$) or slow fading ($\bar{\rho} = 0.99$). To make numerical comparisons, we consider the difference in SNR (in dB) of two schemes to achieve a throughput of 2 bit/symbol.

2.6.0.1 No Collision, Slow Fading

From Fig. 2.7, the maximum achievable throughput $\mathcal{T}^*_{\text{delayed}}$ for delayed CSI incurs an SNR loss of around 2 dB compared to $\mathcal{T}^*_{\text{full}}$ for full CSI. Moreover, $\mathcal{T}^*_{\text{delayed}}$ serves as an upper bound for $\mathcal{T}^*_{\text{ACK-rate}}$ which cannot be directly computed. We also see that both new upper bounds $\mathcal{T}^*_{\text{periodic}}, \mathcal{T}_{\text{ub}}$ are tighter than $\mathcal{T}^*_{\text{delayed}}$ by about 0.2 dB.

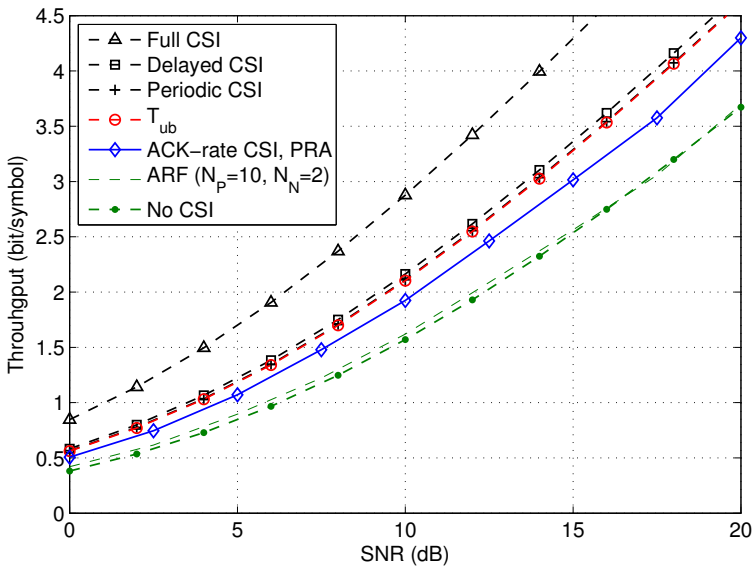


Fig. 2.8: The PRA compared to the benchmarks and upper bound. Parameters: $\bar{\rho} = 0.95$, $q_{10} = 0$, $q_{01} = 1$.

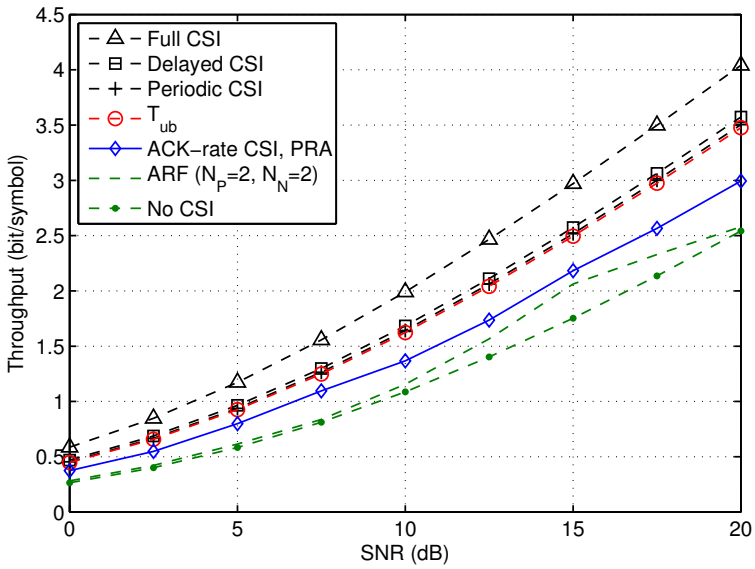


Fig. 2.9: The PRA compared to the benchmarks and upper bound. Parameters: $\bar{\rho} = 0.99$, $q_{10} = 0.4$, $q_{01} = 0.9$.

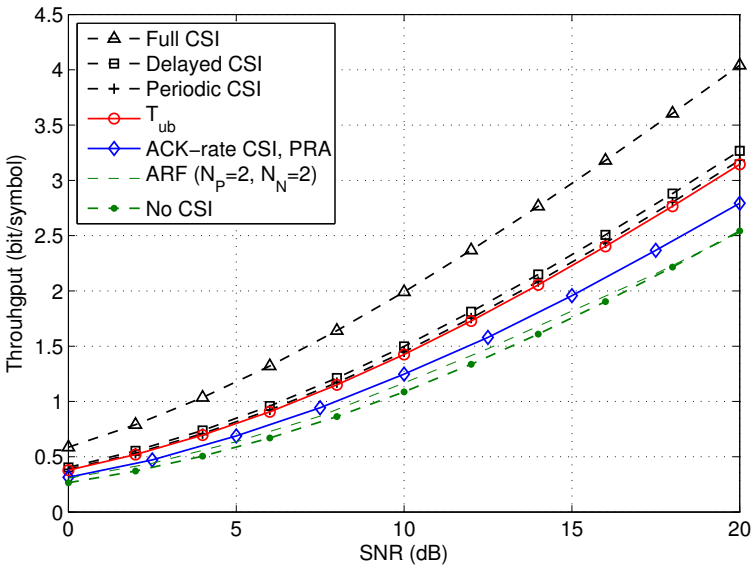


Fig. 2.10: The PRA compared to the benchmarks and upper bound. Parameters: $\bar{\rho} = 0.95$, $q_{10} = 0.4$, $q_{01} = 0.9$.

Fig. 2.7 also shows that the myopic policy with ACK-rate CSI, implemented by PRA with $L = 0$, is about one dB away from the tightest upper bound. This implies that the myopic policy cannot be more than one dB away from the *optimal policy* for ACK-rate CSI. Furthermore, we see that using $L = 1$, compared to using $L = 0$, improves the performance by a few tenths of dB at some SNRs. At low SNR, in particular, the performance becomes very close to the upper bound. Increasing L moderately (say $L = 2, 3, 4$) is not likely to bring about significant gain, but the complexity would already be prohibitive. Subsequently, we focus on myopic policy where $L = 0$.

Generally, the maximum achievable throughput $\mathcal{T}_{\text{no}}^*$ for no CSI serves as a benchmark. We also consider the auto-rate fallback (ARF) scheme [21] as another benchmark. In the ARF scheme, the rate is increased after N_{P} consecutive PACKS and decreased after N_{N} consecutive NACKs. Note that the ARF scheme uses only past ACKs, but not past rates, as CSI. We conducted a search by simulations to optimize $N_{\text{P}}, N_{\text{N}}$ (the optimized values are shown in the legend). Incidentally, the optimized values of $N_{\text{P}} = 10, N_{\text{N}} = 2$ are the same as those considered in [21]. From Fig. 2.7, we observe that the PRA that exploits ACK-rate CSI performs significantly better than both benchmarks. For example, the PRA requires at least 3.1 dB less SNR at a throughput of 2 bit/symbol compared to $\mathcal{T}_{\text{no}}^*$. Some rate adaptation schemes achieves higher throughput with more extensive or more informative CSI, e.g., [56–58] which exploit the RTS/CTS mechanism, but they consume more channel resources than ACK feedback and so may not be suitable as benchmarks.

Finally, we investigate the degradation of using a smaller set of rates for adaptation,

Scenario	Gap to upper bound $\mathcal{T}_{\text{ub}}$	Gap to benchmark $\mathcal{T}^*_{\text{no}}$
No Collision, Slow Fading	1 dB (0.2 dB)	3 dB
No Collision, Moderate Fading	1 dB (0.3 dB)	2 dB
Collisions, Slow Fading	1.75 dB (0.4 dB)	2.6 dB
Collisions, Moderate Fading	1.5 dB (0.4 dB)	1.5 dB

Table 2.2: Summary of the performance of PRA using myopic optimization, in terms of the difference in SNR to achieve a throughput of 2 bit/symbol. The improvement in the upper bound $\mathcal{T}^*_{\text{delayed}} - \mathcal{T}_{\text{ub}}$ is given within the brackets.

as available in IEEE 802.11a, i.e., $\mathcal{S}_R = \{0.5, 0.75, 1, 1.5, 2, 3, 4, 4.5\}$ bit/symbol. The channel state space $\mathcal{S}_H$ remains unchanged. We use a myopic policy with $L = 0$. Fig. 2.7 shows that the performance has degraded[6] by about 1 dB at low SNR. At high SNR, the degradation becomes more significant. This suggests that, from the perspective of rate adaptation, the throughput of IEEE 802.11a may be further improved, especially at high SNR, by increasing the set of available rates.

2.6.0.2 No Collision, Moderate Fading

Fig. 2.8 shows that the throughput typically reduces when the speed of fading increases. Clearly, $\mathcal{T}^*_{\text{full}}$ is not affected as full CSI is available. However, $\mathcal{T}^*_{\text{delayed}}$ now incurs an additional SNR loss of around 1 dB compared to slow fading. This fundamental loss results from the temporal variation of the channel and the causality constraint imposed in practice, and is irrecoverable. We observe that both the proposed upper bounds $\mathcal{T}^*_{\text{periodic}}$ and $\mathcal{T}_{\text{ub}}$ are tighter than $\mathcal{T}^*_{\text{delayed}}$ by about 0.3 dB.

2.6.0.3 Collisions, Slow and Moderate Fading

From Fig. 2.9 (slow fading) and Fig. 2.10 (moderate fading), the throughput is generally further reduced due to collisions. According to Theorem 2.1, this reduction is $p(\epsilon = 0) \approx 30\%$ for $\mathcal{T}^*_{\text{full}}$, $\mathcal{T}^*_{\text{delayed}}$, $\mathcal{T}^*_{\text{no}}$. The new upper bound $\mathcal{T}_{\text{ub}}$ tightens $\mathcal{T}^*_{\text{delayed}}$ by about 0.4 dB.

The performance of the myopic policy with ACK-rate CSI, i.e., the PRA with $L = 0$, is summarized in Table 2.2. We compare the PRA to the tightest upper bound $\mathcal{T}_{\text{ub}}$ and to the benchmark $\mathcal{T}^*_{\text{no}}$, and also show the amount of tightening of the upper bound. Although this PRA is only optimal in a myopic sense, it is within $1 - 1.5$ dB to the maximum achievable throughput and can improve performance by at least 1.5 dB, at a throughput of 2 bit/symbol. These results suggest that the PRA may be a good pragmatic approach for rate adaptation, especially for slowly fading channels with a low probability of collision.

[6]The upper bounds and the benchmarks would also degrade accordingly. We omit the corresponding graphs for clarity of presentation.

2.7 Discussion

We have approached the rate adaptation problem by treating the collision probability as a fixed exogenous variable. Practically, this probability is a endogenous variable, since it depends on the buffer queues, channel fades and also the rates used in a dynamic manner. Nevertheless, our work offers insights on the difficulty of the rate adaptation problem and possible practical solutions. Moreover, we may treat the communication system as having an approximately fixed collision probability over a short period of time and apply our results in such a quasi-stationary setting.

To achieve high throughput, the outage probability is typically high, say $10\% - 30\%$. This high error probability is inappropriate for delay-sensitive applications. To provide a guaranteed quality of service, the rate adaptation problem may be alternatively stated as the maximization of throughput subject to a maximum outage probability $p_{\max}$. Hence, our problem where $p_{\max} = 1$ is a special case of this generalized setting.

In our study we have assumed that the parameters that described the long-term channel statistics are fixed and known. In practice, the parameters may not be exactly known or may even fluctuate over time. This necessitates an online estimation of the parameters which can be another interesting direction for further research.

For future research, the formulations and approaches used in this chapter may be extended to other types of CSIs, e.g., when an additional quantized feedback of the channel state is available, so as to account for other feedback schemes used in practice.

2.8 Conclusion

We have considered packet-by-packet rate adaptation to improve the average throughput over an infinite-time horizon, based on past ACKs and past rates as partial channel state information. We have taken collisions into account in our Markovian channel. Since the maximum achievable throughput cannot be practically computed, we have proposed two new upper bounds. We have shown that the myopic policy, which maximizes only the current throughput, achieves a throughput that is within one dB of the tightest upper bound over a wide range of SNRs for a slowly time-varying channel. This result suggests that the myopic policy is already fairly close to the maximum achievable throughput, yet at a reasonable complexity. Further, the particle filter is proposed to maintain the belief states necessary for throughput optimization. By using the particle-filter-based rate adaptation (PRA), observations on the rate adaptation behavior were made without setting up any *a priori* constraints on the set of rates used, which can be pruned subsequently to reduce the required number of rates.

Appendix 2.A An Auxiliary Lemma

We state an auxiliary lemma that will be used in the proof of Lemma 2.5 and Theorem 2.3.

Lemma 2.3. *Consider random variables $x, y, \widetilde{H}$ which form a Markov chain $x \to y \to \widetilde{H}$, i.e., $p(x, y, \widetilde{H})$ can be factorized as $p(x)p(y|x)p(\widetilde{H}|y)$. Then, the maximum expected value of an objective function $f(R, \widetilde{H})$ obtained by optimizing R is larger given y than given x, i.e.,*

$$\mathbb{E}_y \left[\max_R \mathbb{E}_{\widetilde{H}|y}[f(R, \widetilde{H})] \right] \geq \mathbb{E}_x \left[\max_R \mathbb{E}_{\widetilde{H}|x}[f(R, \widetilde{H})] \right]. \tag{2.34}$$

Proof of Lemma 2.3. The operator $\mathbb{E}_{\widetilde{H}|x}$ is equivalent to $\mathbb{E}_{y|x}\mathbb{E}_{\widetilde{H}|x,y}$, while $\mathbb{E}_{\widetilde{H}|x,y}$ can be replaced by $\mathbb{E}_{\widetilde{H}|y}$ due to $x \to y \to \widetilde{H}$. Thus, the R.H.S of (2.34) can be written as

$$\begin{aligned} \mathbb{E}_x \left[\max_R \mathbb{E}_{y|x}\mathbb{E}_{\widetilde{H}|y} \left[f(R, \widetilde{H}) \right] \right] &\leq \mathbb{E}_x \mathbb{E}_{y|x} \left[\max_R \mathbb{E}_{\widetilde{H}|y} \left[f(R(x), \widetilde{H}) \right] \right] \\ &= \mathbb{E}_y \left[\max_{R(y)} \mathbb{E}_{\widetilde{H}|y} \left[f(R(y), \widetilde{H}) \right] \right]. \end{aligned}$$

The inequality arises from interchanging the $\max_R$ and $\mathbb{E}_{y|x}$ operators, since

$$\max_R \sum_i p_i g(R) \leq \sum_i p_i \max_R g(R)$$

for non-negative p_i and some function g. This thus proves (2.34). □

Appendix 2.B Proof of Theorem 2.2

Lemmas 2.4, 2.5, 2.6 below provide an ordering of the maximum achievable throughput for different CSIs, hence establishing Theorem 2.2.

Lemma 2.4. *Let $\mathcal{T}^*_{\text{any}}$ be the maximum achievable throughput for any type of CSI. Then, $\mathcal{T}^*_{\text{full}} \geq \mathcal{T}^*_{\text{any}} \geq \mathcal{T}^*_{\text{no}}$ and consequently, $\mathcal{T}^*_{\text{full}} \geq \mathcal{T}^*_{\text{delayed}} \geq \mathcal{T}^*_{\text{no}}$.*

Proof. Clearly $T(R_k; C_k) = \mathbb{E}_{\widetilde{H}_k|C_k} \left[t(R_k, \widetilde{H}_k) \right] \leq \mathbb{E}_{\widetilde{H}_k|C_k} \left[\max_{R_k} t(R_k, \widetilde{H}_k) \right]$ for any CSI C_k and rate R_k. Taking the expectation over C_k, we get

$$\mathbb{E}_{C_k} \left[T(R_k; C_k) \right] \leq \mathbb{E}_{\widetilde{H}_k} \left[\max_{R_k} t(R_k, \widetilde{H}_k) \right] = \mathcal{T}^*_{\text{full}}$$

due to (2.13a). From (2.11), we thus get $\mathcal{T}(\pi) \leq \mathcal{T}^*_{\text{full}}$ for any policy π given any CSI. Then, maximizing $\mathcal{T}(\pi)$ over all policies establishes $\mathcal{T}^*_{\text{any}} \leq \mathcal{T}^*_{\text{full}}$. To see that $\mathcal{T}^*_{\text{any}} \geq \mathcal{T}^*_{\text{no}}$, we note that a policy given any CSI can always choose not to use the CSI and achieve $\mathcal{T}^*_{\text{no}}$, hence using the CSI optimally can only achieve the same or greater throughput. This completes the proof. □

We say a CSI C_k at time k is *causal* if $C_k \to \widetilde{H}_{k-1} \to \widetilde{H}_k$. Thus, ACK-rate CSI, delayed CSI, periodic CSI and no CSI are all causal.

Lemma 2.5. *The throughput achieved by a myopic policy with causal CSI C_k is not more than $\mathcal{T}^*_{\text{delayed}}$, i.e.,*

$$\mathbb{E}_{C_k}\left[\max_R T(R; C_k)\right] \le \mathcal{T}^*_{\text{delayed}}. \tag{2.35}$$

Moreover, we have $\mathcal{T}^*_{\text{delayed}} \ge \mathcal{T}^*_{\text{periodic}}$.

Proof. We know from Theorem 2.1 that $\mathcal{T}^*_{\text{delayed}}$ is achieved by a myopic policy. Since $C_k \to \widetilde{H}_{k-1} \to \widetilde{H}_k$, (2.35) follows immediately by applying Lemma 2.3 in Appendix 2.A by defining the objective function f as the throughput t in (2.7), and random variables $x, y, \widetilde{H}$ as $C_k, \widetilde{H}_{k-1}, \widetilde{H}_k$, respectively. To show $\mathcal{T}^*_{\text{periodic}} \le \mathcal{T}^*_{\text{delayed}}$, from the definition (2.21) with periodic CSI $\widehat{C}_k$ we get

$$\mathcal{T}^*_{\text{periodic}}(\pi) = \frac{1}{P}\max_{\pi}\mathbb{E}\left[\sum_{k=1}^{P} T(R_k; \widehat{C}_k)\right] \le \frac{1}{P}\mathbb{E}\left[\sum_{k=1}^{P}\max_{R_k} T(R_k; \widehat{C}_k)\right]. \tag{2.36}$$

Here, the inequality arises from interchanging the $\max_\pi$ and $\mathbb{E}$ operators. We have also replaced $\max_\pi$ by $\max_R$ on the R.H.S. of (2.36), since the rate R_k in π now optimizes each T separately given $\widehat{C}_k$. Finally, since $\widehat{C}_k$ is causal, by applying (2.35) we obtain $\mathcal{T}^*_{\text{periodic}} \le \mathcal{T}^*_{\text{delayed}}$. □

Lemma 2.6. $\mathcal{T}^*_{\text{periodic}} \ge \mathcal{T}^*_{\text{ACK-rate}}$.

Proof. In addition to the ACK-rate CSI, the periodic CSI is given by the periodic update of a delayed channel amplitude at $k \in \mathcal{S}_P$. A policy given periodic CSI can always choose not to use this additional CSI and yet achieves $\mathcal{T}^*_{\text{ACK-rate}}$. Using the periodic CSI optimally can only achieve the same or greater throughput, hence $\mathcal{T}^*_{\text{periodic}} \ge \mathcal{T}^*_{\text{ACK-rate}}$. □

Appendix 2.C Proof of Theorem 2.3

We note that to obtain $\mathcal{T}_{\text{ub}}$, the CSI used is causal and moreover only the throughput of the current packet k is maximized. Hence, Lemma 2.5 applies. From (2.35), we thus obtain $\mathcal{T}^*_{\text{delayed}} \ge \mathcal{T}_{\text{ub}}$. We now show that $\mathcal{T}_{\text{ub}} \ge \mathcal{T}^*_{\text{ACK-rate}}$. From (2.11) and (2.12), we can write

$$\mathcal{T}^*_{\text{ACK-rate}} = \lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}_{C^*_k}\left[T(R^*_k(C^*_k); C^*_k)\right] \tag{2.37}$$

where R^*_k denotes the rate selected by the optimal deterministic policy π^*, and

$$C^*_k = \{R^*_1, \cdots, R^*_{k-1}, A^*_1, \cdots, A^*_{k-1}\}$$

denotes the ACK-rate CSI that has a distribution resulting from the use of π^*. Each summand can be bounded as

$$\begin{aligned}
\mathbb{E}_{C_k^*}\left[T(R_k^*(C_k);C_k)\right] &= \mathbb{E}_{C_k^*}\left[\mathbb{E}_{\widetilde{H}_k|C_k^*}\left[t(R_k^*(C_k^*);\widetilde{H}_k)\right]\right] && \text{(2.38a)}\\
&\le \mathbb{E}_{C_k^*}\left[\max_{R_k}\mathbb{E}_{\widetilde{H}_k|C_k^*}\left[t(R_k;\widetilde{H}_k)\right]\right] && \text{(2.38b)}\\
&\le \mathbb{E}_{\bar{C}_k^*}\left[\max_{R_k}\mathbb{E}_{\widetilde{H}_k|\bar{C}_k^*}\left[t(R_k;\widetilde{H}_k)\right]\right] && \text{(2.38c)}\\
&= \mathbb{E}_{\widetilde{H}_{k-2}}\mathbb{E}_{R_{k-1}^*,A_{k-1}^*|\widetilde{H}_{k-2}}\max_{R_k}T(R_k;\bar{C}_k^*) && \text{(2.38d)}\\
&\le \mathbb{E}_{\widetilde{H}_{k-2}}\max_{R_{k-1}}\mathbb{E}_{A_{k-1}|\widetilde{H}_{k-2}}\left[\max_{R_k}T(R_k;\bar{C}_k)\right] && \text{(2.38e)}\\
&= \mathcal{T}_{\mathrm{ub}}. && \text{(2.38f)}
\end{aligned}$$

Here, (2.38a) follows from the definition (2.8); (2.38b) follows from replacing $R_k^*(C_k)$ as variable R_k to be optimized given C_k; (2.38c) follows from Lemma 2.3 in Appendix 2.A, since it can be shown that $C_k^* \to \bar{C}_k^* \to \widetilde{H}_k$, where we define $\bar{C}_k^* \triangleq \{R_{k-1}^*, A_{k-1}^*, \widetilde{H}_{k-2}\}$; (2.38d) follows directly from the definition of $\bar{C}_k^*$ and (2.8). Next, (2.38e) follows from replacing (R_{k-1}^*, A_{k-1}^*) as a pair of variables (R_{k-1}, A_{k-1}) to be optimized given $\widetilde{H}_{k-2}$, but since A_{k-1} depends only on R_{k-1} (given $\widetilde{H}_{k-2}$), it is sufficient to optimize only R_{k-1}. Thus, $\bar{C}_k^*$ in (2.38d) is replaced by $\bar{C}_k = \{R_{k-1}, A_{k-1}, \widetilde{H}_{k-2}\}$ in (2.38e). Finally, (2.38f) follows from the definition (2.16). It follows that each summand in (2.37) is no greater than $\mathcal{T}_{\mathrm{ub}}$. Thus, $\mathcal{T}_{\mathrm{ACK-rate}}^* \le \mathcal{T}_{\mathrm{ub}}$. This concludes the proof.

CHAPTER 3

INCREMENTAL-REDUNDANCY INCREMENTAL-DATA ARQ CODING SCHEME

We consider communication systems where data is transmitted as packets of fixed (large) length. In such systems, incremental redundancy (IR) is typically sent in retransmissions to recover erroneous packets. In IR, the entire packet is used for sending redundancy, so as to correct an erroneous packet. This constitutes a waste of channel resources, if a small amount of IR is already sufficient for packet recovery. In this chapter, in addition to IR (usually small in amount), we propose to send incremental data (ID). This ID contains new information bits that are appropriately coded but are not sent previously. We call this ARQ scheme the incremental-redundancy incremental-data coding (IRIDC) scheme.

To illustrate the benefit of this scheme, we consider the problem of maximizing the throughput over a frame of K packets, assuming a block-fading channel where the channel changes independently after every frame. A lean feedback, in the form of acknowledgement bits, is available at the transmitter as channel state information. We show that the optimizing problem can be solved by dynamic programming, but the optimal rate-adaptation policy is still difficult to compute practically. To simplify the computation, we impose the equal-rate constraint (ERC), where the ID are encoded at the same rate as the effective rate of the IR. Numerical results in Rayleigh fading channels show that substantial throughput improvement can be achieved using the IRIDC scheme under the ERC, compared to conventional ARQ schemes.

3.1 Introduction

Most current and future wireless communication systems are packet switched, such as wireless LANs that follow the IEEE 802.11 standards [9, 15], and cellular communications that follow 3G LTE [33]. In a packet-switched system, the transmitter sends information to a receiver in blocks of data called packets. Automatic repeat request (ARQ) [6,7] is often used in the data link control sublayer to regulate retransmissions from a transmitter to a receiver. For this purpose, an acknowledgement (ACK) bit is sent by the receiver to indicate whether the packet is received successfully, in the form of a positive ACK (PACK) or a negative ACK (NACK). Typically, a missing PACK is also interpreted as a NACK. The ACKs can be considered as a form of partial channel state information (CSI) to the transmitter, which can be used to improve its transmission strategy.

ARQ systems can be classified as Type I and Type II hybrid ARQ schemes. In Type I hybrid ARQ schemes, failed transmissions are discarded and not used to improve the decoding of the failed information bits. In Type II hybrid ARQ schemes, failed transmissions are instead saved in a buffer at the receiver. These failed transmissions are then used jointly with subsequent retransmissions in decoding the data symbols. Since joint decoding improves the probability of successfully recovering the data, the throughput is also improved. The redundancy bits in the retransmissions can be sent incrementally in the form of so-called *incremental redundancy* (IR) bits, which are usually significantly smaller in number than the bits sent in the failed transmission. Typically, IR is continually sent in small blocks until the data symbols are decoded successfully, after which a PACK is sent to the transmitter. At this point, a just-sufficient amount of redundancy has been used to successfully decode the data. Hence, channel resources are used efficiently with IR. A well-known IR code is the rate compatible punctured convolutional code (RCPC) [28]. More advanced capacity-approaching IR codes based on turbo codes [29] and low-density parity-check codes [30] have also been designed.

In some communication systems, blocks of channel resources called slots, each of a fixed length, are reserved for transmissions for each user [69, 70]. Every packet is then transmitted in one of these (fixed-length) slots. In [69, 70], a Type II hybrid ARQ scheme is used. Specifically, to recover previously-failed packets, an entire slot is used to send redundancy bits, and so the effective code rate of the transmissions is fixed at $R/(n+1)$, where R is the code rate of the original transmission and n is the number of retransmissions. However, for some channel scenarios, only a small incremental amount of redundancy is necessary for successful decoding of the data. In such cases, by employing the entire slot for only transmitting redundancy, more channel resources than necessary are used in the retransmission. The excess channel resources could have been used instead to transmit new information bits.

To introduce flexibility into the Type II hybrid ARQ scheme, in this chapter we propose to send *incremental data* (ID), in addition to IR, in retransmissions. The ID are new information bits that are not previously sent and may be appropriately coded. We call this ARQ scheme the incremental-redundancy incremental-data coding

(IRIDC) scheme. In contrast to [69, 70], the IR here consumes a flexible amount of channel resources. The remaining channel resources not used for IR are then used to send ID. This is possible if the source always contains sufficient bits ready for transmission, such as in applications like multimedia streaming. In our proposed ARQ scheme, we time multiplex the IR and ID in the same slot for retransmission, resulting in a simple implementation. Another approach to combine the IR and ID is to send both concurrently using superposition codes. However, we show that the time-multiplexing scheme is sufficient in achieving the highest possible rate for Gaussian codes[1].

To illustrate the potential benefit of the IRIDC scheme, we maximize the throughput of a slot-based communication system. A channel state information (CSI) is provided at the transmitter in form of an ACK, after every packet is sent. We consider a block-fading channel in which the channel is quasi-static over a frame of K packets, but changes independently after every frame[2]. To maximize throughput, we show that dynamic programming can be used to determine the optimal rate-adaptation policy, by optimally choosing the parameters used for IR and ID over K packets.

To simplify the computation of the optimal policy, we propose to encode new information bits at the same rate as the rate of the original information bits taking into account all IR used so far. That is, all information bits are constrained to be encoded at the same *effective rate*. We call this the *equal-rate constraint* (ERC). Numerical results obtained for Rayleigh fading channels show that substantial throughput improvement can be achieved using IRIDC scheme, compared to conventional ARQ schemes. In particular for $K = 4$ and for most practical SNRs (less than 40 dB), the IRIDC scheme, using only ACKs as CSI at the transmitter, achieves a throughput that is within one bit/symbol of the maximum achievable throughput with full channel knowledge.

The chapter is organized as follows. Section 3.2 introduces the system model for ARQ and proposes IRIDC. Then, Section 3.3 elaborates on the IRIDC. As an application for IRIDC, Section 3.4 considers the problem of throughput maximization and provides an optimal solution by dynamic programming. To simplify the optimization, Section 3.5 imposes the ERC. Numerical results are given in Section 3.6. Finally, the conclusion is given in Section 3.8.

3.2 System Model for ARQ

We first introduce the block-fading channel in Section 3.2.1. In Section 3.2.2, we give a general description of the *causal* coding schemes that can be carried out in ARQ systems. Under this causality constraint, we consider some conventional ARQ coding schemes in Section 3.2.3, and we propose IRIDC in Section 3.2.4.

[1] A superposition scheme may however achieve a better error exponent, compared to the time-multiplexing scheme, and hence may be more appropriate for short packets.

[2] This corresponds, for instance, to the scenario where every user is assigned a frame of K slots for transmission, but the frames for the same user are spaced far apart in time.

3.2.1 Block-Fading Channel

We consider a block-fading channel where the channel coefficient $h \in \mathbb{C}$ is fixed over a *frame* of length KN, but can change independently from frame to frame. We assume that any information in the previous frames is discarded and not used, while information within the frame can be kept in memory and exploited.

Henceforth, we shall focus on the transmission of one frame. In each frame, K packets are sent. Each packet is used to transmit N symbols. We assume that N is sufficiently large such that the rates defined in this chapter are achievable for arbitrarily small packet error rates. Let $\mathbf{x}_k \in \mathbb{C}^N$ be the transmitted signal vector in packet k, where $k = 1, \cdots, K$. The received signal vector is given by

$$\mathbf{y}_k = h\mathbf{x}_k + \mathbf{n}_k, k = 1, \cdots, K, \tag{3.1}$$

where $\mathbf{n}_k \in \mathbb{C}^N$ is additive white Gaussian noise distributed as $\mathbf{n}_k \sim \mathcal{CN}(0, 1)$, a circularly symmetric complex independent identical distributed (i.i.d.) Gaussian distribution with zero mean and unit variance. For our simulations and analysis, we focus on Rayleigh-fading channels.

For simplicity, we assume the use of Gaussian codewords with unit power, i.e., $\mathbf{x}_k \sim \mathcal{CN}(0, 1)$. The Gaussian distribution may not necessarily be optimal with ARQ. However, if ARQ is not implemented, we note that for Gaussian channels, where the channel h is fixed for all frames and only AWGN is present, the Gaussian distribution is optimal in achieving the capacity of $\mathsf{C}(\bar{\gamma}) = \log(1 + \bar{\gamma})$, where $\bar{\gamma} = |h|^2$.

3.2.2 Causal Encoding in ARQ Systems

To simplify the process of packet transmission above the physical layer, we neglect the transmission of any overhead. Hence, we treat $\mathbf{x}_k$ equivalently either as a codeword (after encoding) or as a packet (for transmission). At time k, message w_k which contains $N\beta_k$ information bits is encoded as $\mathbf{x}_k$ and sent. The information bits are generated independently and are drawn from a source which, for simplicity, is assumed to always contain sufficient bits. In addition, we allow past messages within the frame that have not been recovered so far to be used as inputs for encoding packet k.

After packet k is transmitted, a single-bit ACK A_k is fed back to the transmitter indicating whether past packets within the frame are received correctly. The precise definition for NACK and PACK depends on the ARQ scheme and will be given later. The consolidated feedback from initial time 1 onwards is written as an ACK vector $\mathbf{a}_k = [A_1, \cdots, A_k]$. For convenience, we define $\mathbf{a}_0 = \emptyset$, a null value.

Depending on the past ACKs, we decide which past and present messages to use as inputs for encoding the present codeword. For full generality, we assume that all past and present messages may be used. The codeword is then expressed as

$$\mathbf{x}_k = f(w_1, \cdots, w_k; \mathbf{a}_{k-1}), \tag{3.2}$$

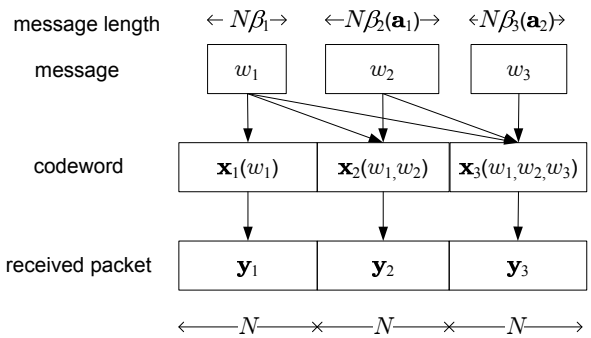


Fig. 3.1: System model for coding in ARQ systems. At time k, a codeword $\mathbf{x}_k$ is encoded based on present message w_k and past messages $w_1, \cdots, w_{k-1}$, as well as based on past ACKs $\mathbf{a}_{k-1} = [A_1, \cdots, A_{k-1}]$. Then, the codeword is sent over a memoryless channel and received as $\mathbf{y}_k$.

where $f : \mathbb{B}^{N\beta_1} \times \cdots \times \mathbb{B}^{N\beta_k} \rightarrow \mathbb{R}^N$ is the encoding function with parameter $\mathbf{a}_{k-1}$ and $\mathbb{B} = \{0, 1\}$. The lengths of the past messages have been fixed, but if the present message is used for encoding, its length $N\beta_k = N\beta_k(\mathbf{a}_{k-1})$ depends on the past ACKs. Fig. 3.1 shows the Markovian dependence of the message, codeword and received packet over time for a frame of $K = 3$ packets.

3.2.3 Known ARQ Schemes

We describe two well-known coding schemes, both of which fall under the causal encoding strategy described by (3.2). Then, we explain how rate adaptation can be performed for these schemes.

3.2.3.1 Independent coding (IC)

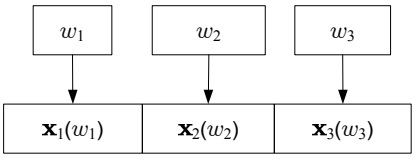
In the IC scheme, each message is independently encoded regardless of past ACKs (even when an NACK is received). This is also known as a Type I hybrid ARQ scheme in the literature. Specifically, the codeword is encoded as

$$\mathbf{x}_k = f(w_k; \mathbf{a}_{k-1}) \tag{3.3}$$

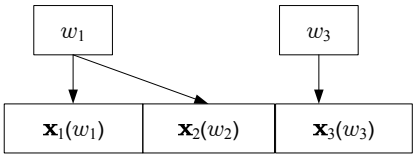
for any $\mathbf{a}_{k-1}$. Since only a new message is encoded regardless of the past ACKs, a PACK is used only to indicate that the last-received packet is decoded successfully:

$$A_k = 1 \Leftrightarrow \{w_k \text{ is successfully recovered}\}. \tag{3.4}$$

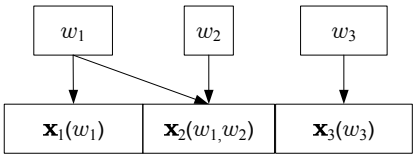
Fig. 3.2(a) shows the relationship of the message and the codeword over time for $K = 3$. The dependence of the codewords on the messages are indicated by arrows. For IC, we see in Fig. 3.2(a) that each codeword $\mathbf{x}_k$ depends only on the message w_k, according to (3.3).



(a) Independent coding (IC). A new message is always transmitted; past messages are always discarded.



(b) Incremental-redundancy coding (IRC). A new message is transmitted and past messages are discarded if and only if past messages are received successfully.



(c) Incremental-redundancy incremental-data coding (IRIDC). A new message is always transmitted; past messages are discarded if received successfully.

Fig. 3.2: Three causal coding schemes, illustrated for the case when the past ACKs are realized as $\mathbf{a}_2 = [0, 1]$.

3.2.3.2 Incremental-redundancy coding (IRC)

In IRC, if a PACK is received, a new message is sent. This is similar to IC. If a NACK is received, however, redundancy based on the same erroneous message is sent continually, until the erroneous message is successfully recovered; in contrast to IC, a new message is *not* sent. This scheme is known as Type II hybrid ARQ scheme in the literature [69, 70]. Typically, IRC offers stronger error protection than IC and hence increases the reliability of data transmissions. Specifically, the codewords are encoded as

$$\mathbf{x}_k = \begin{cases} f(w_k; \mathbf{a}_{k-1}), & A_{k-1} = 1 \text{ or } k = 1; \\ f(w_{\tilde{k}}; \mathbf{a}_{k-1}), & A_{k-1} = 0, \end{cases} \tag{3.5}$$

where $w_{\tilde{k}}$ refers to the last erroneous message. Here, $k = 1$ refers to the first transmission in the frame, hence past packets are discarded and a new packet is always sent (this applies for all ARQ schemes). Further, in IRC a PACK is sent according to

$$A_k = 1 \Leftrightarrow \{w_{\tilde{k}} \text{ are successfully recovered}\}. \tag{3.6}$$

Fig. 3.2(b) shows the relationship of the message and the codeword over time for the example of $\mathbf{a}_2 = [0, 1]$. For IRC, we see in Fig. 3.2(b) that besides codeword $\mathbf{x}_1$, codeword $\mathbf{x}_2$ also depends on message w_1, since $A_1 = 1$ and so packet 1 has not yet been recovered. After packet 2 is received correctly (since $A_2 = 1$), w_1 and w_2 can then be discarded and now $\mathbf{x}_3$ depends only on w_3.

3.2.3.3 Rate Adaptation for IC and IRC

Rate adaptations, based on input of ACKs $\mathbf{a}_k$, can be carried out for both IC and IRC by varying the message size $\beta_k(\mathbf{a}_k)$. In IC, the rate can be flexibly adapted for any packet k by varying β_k. In IRC, if a PACK is received, the subsequent rate can also be flexibly adapted. However, if an NACK is received, the rate is fixed as $R/(n+1)$ for the nth retransmission, where R is the rate used by the message that is yet to be recovered.

3.2.4 Incremental-Redundancy Incremental-Data Coding

We will propose incremental-redundancy incremental-data coding (IRIDC) for ARQ by specializing the causal encoding strategy (3.2). We shall show that IC and IRC are special cases of IRIDC under certain constraints.

In IRIDC, both IR and ID are sent as long as previous packets are not decoded successfully. The motivation for this approach is that in some moderately good channels, even though a packet is received erroneously, only a relatively small amount of IR is needed to recover the failed packets. The remaining packet resources can hence be used for sending additional ID (coded with appropriate amount of redundancy) to increase the throughput of the system.

Specifically in IRIDC, an ACK bit, in the form of a PACK $A_k = 1$ or NACK $A_k = 0$, is used to indicate whether *all past messages* are decoded correctly, i.e.,

$$A_k = 1 \Leftrightarrow \{w_1, \cdots, w_k \text{ are successfully recovered}\}, \tag{3.7}$$

Consequently, when a NACK first occurs, a continual burst of NACKs will build up until all packets are successfully decoded or until the packet index exceeds the frame length.

The IRIDC scheme can now be specified as follows. If $k = 1$, clearly a new message is sent. Consider $k \geq 2$. If the previous packet is received successfully, i.e., $A_{k-1} = 1$, then a new message is sent. The distinction of the IRIDC scheme with other schemes lies mainly in the case when $A_{k-1} = 0$. In this case, both IR (based on the erroneous messages) and incremental data (based on the new message) will be sent. Mathematically, we thus have

$$\mathbf{x}_k = \begin{cases} f(w_k; \mathbf{a}_{k-1}), & A_{k-1} = 1 \text{ or } k = 1; \\ f(w_{\tilde{k}}, \cdots, w_k; \mathbf{a}_{k-1}), & A_{k-1} = 0. \end{cases} \tag{3.8}$$

where $\tilde{k}$ denotes the index of the first packet in the last burst of NACKs.

Both IC and IRC are special cases of IRIDC. We obtain (3.3) for IC by dropping the dependence of $w_{\tilde{k}}, \cdots, w_{k-1}$ when $A_{\tilde{k}-1} = 0$ in (3.8). Also, we obtain (3.5) for IRC by dropping the dependence of $w_{\tilde{k}+1}, \cdots, w_k$ when $A_{\tilde{k}-1} = 0$ in (3.8).

Fig. 3.2(c) illustrates an example of the relationship of the messages and the codewords over time when $\mathbf{a}_2 = [0, 1]$. As shown in Fig. 3.2(c), packet 1 sends coded bits based

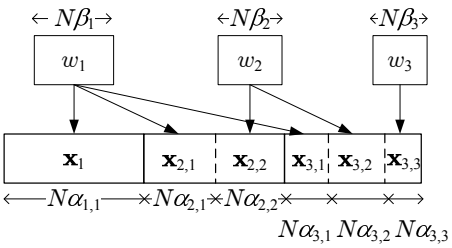


Fig. 3.3: IRIDC by time-division multiplexing the incremental redundancy and incremental data. For packet k, the parameters $\mathcal{P}_k = \{\beta_k, \alpha_{k,i}, i = 1, \cdots, k\}$ can be adapted, given past ACKs $\mathbf{a}_{k-1} = [A_1, \cdots, A_{k-1}]$.

on message w_1, but this message is not received correctly with $A_1 = 0$. In the IRIDC scheme, packet 2 sends IR based on w_1 and also ID based on w_2. Both messages are then received correctly as $A_2 = 1$. Packet 3 then proceeds to send coded bits based only on w_3.

3.3 IRIDC Scheme based on Time-Multiplexing

To illustrate the IRIDC scheme, let us consider time instant k when $\mathbf{a}_{k-1}$ is known, but before packet k is sent. Without loss of generality, suppose that a recent burst of NACKs of length $k - \tilde{k}$ has occurred, where $1 \leq \tilde{k} \leq k$. That is, $\mathbf{a}_{k-1}$ becomes

$$\mathbf{a}_{k-1} = [0, \cdots, 0]$$

for $\tilde{k} = 1$ and

$$\mathbf{a}_{k-1} = [\times, 1, 0, \cdots, 0]$$

for $\tilde{k} = 2, \cdots, k$, where $\times$ denotes arbitrary ACK values, of length $\tilde{k} - 2$. This means that the messages $\{w_i\}$, where the index i runs from $\tilde{k}$ to $k - 1$, have failed and have not yet been decoded successfully; for these messages we will send IR. Moreover, we will send ID to deliver a new message w_k.

We propose to use time-division multiplexing to combine the IR and ID, as shown in Fig. 3.3. Packet k is encoded by time-multiplexing $k - \tilde{k} + 1$ component codewords according to[3]

$$\mathbf{x}_k = \left[\mathbf{x}_{k,\tilde{k}}, \cdots, \mathbf{x}_{k,k}\right], \tag{3.9}$$

Here, the ith component codeword $\mathbf{x}_{k,i}, i = \tilde{k}, \cdots, k$, depends only on message w_i. We consider deterministic encoding, hence we can write the ith component codeword as a function of the message w_i, i.e.,

$$\mathbf{x}_{k,i} = f_{k,i}(w_i),$$

[3]We assume all vectors are row vectors for notational ease.

where $f_{k,i} : \mathbb{B}^{N\beta_i} \to \mathbb{C}^{N\alpha_{k,i}}$ is the encoding function, and $N\alpha_{k,i}$ is the length of $\mathbf{x}_{k,i}$ that may be varied for rate adaptation. Without loss of generality, in (3.9) we use the first $k-\tilde{k}$ component codewords for sending IR and the last component codeword $\mathbf{x}_{k,k}$ for sending ID.

To facilitate discussions, we make the following definitions. Let $\mathcal{S}_k \triangleq \{\tilde{k}, \cdots, k\}$ be the set of time indices of the yet-to-recover messages, including the present message w_k. We define the *effective rate* of yet-to-recover message w_i at time k, $i \le k$, as

$$R_{k,i} = \frac{\beta_i}{\sum_{j=i}^{k} \alpha_{j,i}}, \quad i \in \mathcal{S}_k. \tag{3.10}$$

This effective rate reflects the effective channel utilization of w_i with ARQ. This is because w_i contains $N\beta_i$ information bits while in total $N\sum_{j=i}^{k}\alpha_{j,i}$ symbols are used for its first transmission and subsequent retransmissions.

Henceforth, we assume the use of Gaussian codes that satisfy (3.11) in Lemma 3.1. Lemma 3.1 also shows that time multiplexing the IR and ID is optimal in achieving the maximum possible rate.

Lemma 3.1. *Consider a block-fading Gaussian channel given by (3.1), where the channel coefficient h is fixed over K transmissions. Define $\mathcal{S}_k \triangleq \{\tilde{k}, \cdots, k\}$, where $\tilde{k}$ are fixed*[4] *such that $1 \le \tilde{k} \le k \le K$. Suppose that at time $k = \tilde{k}, \cdots, K$, encoding and decoding are carried out as follows:*

- *encoding according to (3.8): a length-N Gaussian code $\mathbf{x}_k \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$ is used to transmit messages $\{w_i, i \in \mathcal{S}_k\}$,*
- *decoding: the received packets $\{\mathbf{y}_i, i \in \mathcal{S}_k\}$ are used to decode messages $\{w_i, i \in \mathcal{S}_k\}$*[5]*.*

Then, there exists Gaussian codes $\{\mathbf{x}_i, i \in \mathcal{S}_K\}$ such that at time $k = \tilde{k}, \cdots, K$

$$R_{k,i} < \mathsf{C} \Leftrightarrow w_i \text{ is decoded successfully}, \quad i = \tilde{k}, \cdots, k, \tag{3.11}$$

with high probability for all N sufficiently large, where $\mathsf{C} = \log(1+|h|^2)$ is the mutual information for each transmission. Moreover, time-multiplexing according to (3.9), instead of the more general strategy given by (3.8), satisfies $\Rightarrow$ in (3.11).

Proof. Since the mutual information C represents the maximum achievable rate (for Gaussian codes), and $R_{k,i}$ is the effective rate of the transmission for message i, clearly $\Leftarrow$ in (3.11) holds at time $k = \tilde{k}, \cdots, K$ for $i = \tilde{k}, \cdots, k$.

To prove that $\Rightarrow$ in (3.11) holds, we consider a time-multiplexing scheme according to (3.9), which is based more generally on (3.8). For ease of explanation, consider $\tilde{k} = 1$

[4] Although we use $\tilde{k}$ to denote the index of the first packet in the recent burst of NACKs, in this lemma $\tilde{k}$ is fixed but can be arbitrarily chosen subject to $1 \le \tilde{k} \le k$.

[5] For example, at time $\tilde{k}$, only packet $\tilde{k}$ is used to decode $w_{\tilde{k}}$. At time $\tilde{k}+1$, both packets $\tilde{k}, \tilde{k}+1$ are used to decode both $w_{\tilde{k}}, w_{\tilde{k}+1}$. At time K, all packets $\tilde{k}, \cdots, K$ are used to decode $w_{\tilde{k}}, \cdots, w_K$. Hence, this scenario applies to the IRIDC scheme (and also to the IRC scheme) when past packets contain redundancy to assist decoding.

and consider $K = 3$, as shown in Fig. 3.3. According to (3.11), we want to prove that there exists Gaussian codes $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ such that

- At time $k = 1$, by using $\mathbf{y}_1$ for decoding, we have

$$R_{1,1} < \mathsf{C} \Rightarrow w_1 \text{ is decoded successfully.} \tag{3.12a}$$

- At time $k = 2$, by using $\{\mathbf{y}_1, \mathbf{y}_2\}$ for decoding, we have

$$R_{2,1} < \mathsf{C} \Rightarrow w_1 \text{ is decoded successfully,} \tag{3.12b}$$
$$R_{2,2} < \mathsf{C} \Rightarrow w_2 \text{ is decoded successfully.} \tag{3.12c}$$

- At time $k = 3$, by using $\{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3\}$ for decoding, we have

$$R_{3,1} < \mathsf{C} \Rightarrow w_1 \text{ is decoded successfully,} \tag{3.12d}$$
$$R_{3,2} < \mathsf{C} \Rightarrow w_2 \text{ is decoded successfully,} \tag{3.12e}$$
$$R_{3,3} < \mathsf{C} \Rightarrow w_3 \text{ is decoded successfully.} \tag{3.12f}$$

Since time-multiplexing is used, the coding for messages w_1, w_2, w_3 can be considered separately. First, consider message w_1: we want to show that (3.12a),(3.12b),(3.12d) hold. From (3.10), the rates at time $k = 1, 2, 3$ are given by $R_{1,1} = \beta_1/\alpha_{1,1}$, $R_{2,1} = \beta_1/(\alpha_{1,1} + \alpha_{2,1})$ and $R_{3,1} = \beta_1/(\alpha_{1,1} + \alpha_{2,1} + \alpha_{3,1})$, respectively. In [70], it is shown that there exists a mother code consisting of M equal-length component Gaussian codes such that the first $m = 1, \cdots, M$ components are decoded successfully with high probability for large N if $\beta/(m\alpha) < \mathsf{C}$, where $N\beta$ is the number of information bits and $N\alpha$ is the number of symbols used for each component code. By making α to be sufficiently small and M to be sufficiently large, it follows using this result in [70] that if the rates $R_{1,1}, R_{2,1}$ and $R_{3,1}$ are less than the capacity C, the message w_1 can be recovered at time $k = 1, 2, 3$, respectively, with high probability for large N. Hence, it follows that (3.12a),(3.12b),(3.12d) hold. Similarly, it can be shown that with high probability for large N:

- if $R_{2,2} = \beta_2/\alpha_{2,2}$ and $R_{3,2} = \beta_2/(\alpha_{2,2} + \alpha_{3,2})$ are less than the capacity C, the message w_2 can be recovered at time $k = 2, 3$, respectively;
- if $R_{3,3} = \beta_3/\alpha_{3,3}$ is less than the capacity C, the message w_3 can be recovered at time $k = 3$.

Hence, it follows that (3.12b),(3.12c) hold, and also that (3.12f) holds. This shows that $\Rightarrow$ in (3.11) holds for $\tilde{k} = 1$ and $k = 1, 2, 3$.

We note that in the IRIDC scheme specified by (3.8), previous messages that are successfully decoded are not used for encoding. Hence, the encoding and decoding scheme remains unchanged after every PACK. Hence, the proof for $\tilde{k} > 1$ is carried out in the same way as for $\tilde{k} = 1$. Similarly, the proof can be straightforwardly extended for any K. This completes the proof. □

We consider the regime of infinitely large N to simplify our problem, i.e., we take (3.11) to hold exactly, with probability one. Thus, a PACK is sent if and only if for

all $i \in \mathcal{S}_k$, $R_{k,i}$ is less than the capacity C, i.e.,

$$R_k^{\max} < \mathsf{C} \Leftrightarrow A_k = 1, \tag{3.13}$$

where we define the *maximum effective rate* at time k as

$$R_k^{\max} = \max_{i \in \mathcal{S}_k} R_{k,i}. \tag{3.14}$$

From the set of rates $\{R_{k,i}, i \in \mathcal{S}_k\}$ used, we see that only $R_k^{\max}$ is relevant in determining whether a PACK or an NACK is received.

From Lemma 3.1 to the ARQ system, the packets $\tilde{k}, \cdots, k-1$ are not yet recovered. This implies that $\mathsf{C} < R_i^{\max}$ for $i = \tilde{k}, \cdots, k-1$. Such information about the channel capacity can be exploited for throughput maximization, as shown in the next section.

3.4 Throughput Maximization with IRIDC

To illustrate the benefit of IRIDC, we consider the problem of optimizing a rate-adaptation policy, so as to maximize the throughput summed over one frame. For this optimization, we have channel state information (CSI) that consists mainly of past ACKs. We make precise the notions of a CSI, policy and throughput in Sections 3.4.1, 3.4.2, 3.4.3, respectively. Then, we formally state the optimization problem in Section 3.4.4 and obtain a solution by dynamic programming in Section 3.4.5.

Let us first consider the parameters available for optimization. To transmit packet k, we can choose the number of new information bits $N\beta_k$ to send as ID and the number of channel symbols $\{N\alpha_{k,i}, i \in \mathcal{S}_k\}$ to send the IR; see Fig. 3.3. Dividing by N, this (normalized) set of parameters is denoted as

$$\mathcal{P}_k = \{\beta_k, \alpha_{k,i}, i \in \mathcal{S}_k\}, \tag{3.15}$$

where the parameters are subject to the constraints

$$\beta_k \geq 0, \tag{3.16}$$

$$\alpha_{k,i} \geq 0, \tag{3.17}$$

$$\sum_{i \in \mathcal{S}_k} \alpha_{k,i} = 1. \tag{3.18}$$

These constraints ensure that the allocated channel resources are non-negative and that the transmission is fully utilized.

We note that by varying $\mathcal{P}_k$, we are actually varying the rates $\{R_{k,i}, i \in \mathcal{S}_k\}$ and consequently also varying $R_k^{\max}$. In other words, we are performing rate adaptation concurrently for all messages $\{w_i, i \in \mathcal{S}_k\}$ at time k.

3.4.1 Channel State Information (CSI)

To adapt the parameters $\mathcal{P}_k$ at time k, we are given the CSI

$$C_k = \{\mathbf{a}_{k-1}, \mathcal{P}_i(\mathbf{a}_{i-1}), i \in \mathcal{S}_{k-1}(\mathbf{a}_{k-1})\}, k = 2, \cdots, K. \tag{3.19}$$

By definition, $C_k = \emptyset$ for $k = 1$ since no CSI is available. In (3.19), the CSI consists of previous ACKs $\mathbf{a}_{k-1}$ and past parameters $\{\mathcal{P}_i, i \in \mathcal{S}_{k-1}\}$. There is in fact no need to specify these past parameters as CSI, since $\mathcal{P}_i$ depends only on $\mathbf{a}_{i-1}$. Nevertheless, we have included these past parameters to emphasize that they directly provide information for adaptation.

3.4.2 Rate-Adaptation Policy

A rate-adaptation policy $\mathcal{P}$ is defined as the collection of $\mathcal{P}_k(C_k)$ over all C_k and over all k, i.e.,

$$\mathcal{P} = \{\mathcal{P}_k(C_k), \forall C_k, k = 1, \cdots, K\}. \tag{3.20}$$

3.4.3 Throughput

If packet k is received correctly, a total of $\mathcal{B}_k \triangleq \sum_{i \in \mathcal{S}_k} \beta_i$ bits are recovered. This occurs when $A_k = 1$. If $A_k = 0$, a packet lost in an outage is discarded (a common practice in delay-sensitive applications) or retransmitted, but in both cases the throughput is zero. Given CSI C_k, the expected throughput of packet k can therefore be expressed as

$$\begin{aligned} T_k(\mathcal{P}; C_k) &= \mathcal{B}_k \Pr(A_k = 1 | C_k) \\ &= \mathcal{B}_k \Pr(R_k^{\max} \le \mathsf{C} | C_k) \end{aligned} \tag{3.21}$$

by using (3.13). Here, $\mathcal{B}_k, R_k^{\max}$ depend on the policy $\mathcal{P}$. To be more specific, they depend only on the present and past parameters $\{\mathcal{P}_i(\mathbf{a}_{i-1}), i \in \mathcal{S}_k\}$.

We note that the ACKs in the CSI C_k only provide information that the mutual information C lies between a lower and an upper bound, due to (3.13). Let us denote C_{k-1}^{low} and C_{k-1}^{up}, respectively, as lower and upper known bounds for C *after* A_{k-1} is received and C_k is given. Thus, the throughput becomes

$$T_k(\mathcal{P}; C_k) = \mathcal{B}_k \Pr(R_k^{\max} \le \mathsf{C} | C_{k-1}^{\text{low}} \le \mathsf{C} \le C_{k-1}^{\text{up}}). \tag{3.22}$$

For $k = 0$, we let $C_0^{\text{low}} = 0$ and $C_0^{\text{up}} = \infty$ to reflect that no information is available about C. For $k = 1, \cdots, K-1$, we can express $C_k^{\text{low}}, C_k^{\text{up}}$ in a recursive manner as

$$C_k^{\text{low}} = \begin{cases} C_{k-1}^{\text{low}} & A_k = 0; \\ \max\{C_{k-1}^{\text{up}}, R_k^{\max}\} & A_k = 1, \end{cases} \tag{3.23}$$

$$C_k^{\text{up}} = \begin{cases} \min\{C_{k-1}^{\text{low}}, R_k^{\max}\} & A_k = 0; \\ C_{k-1}^{\text{up}} & A_k = 1, \end{cases} \tag{3.24}$$

respectively. To see that (3.23) and (3.24) hold, suppose that $C_{k-1}^{\text{low}} \leq \mathsf{C} \leq C_{k-1}^{\text{up}}$ at an arbitrary time k (which clearly holds for $k = 1$), where $1 \leq k \leq K-1$. If an NACK is received at time k, from (3.13) we know that $\mathsf{C} \leq R_k^{\max}$. If a PACK is instead received at time k, we know that $\mathsf{C} \geq R_k^{\max}$. Combining all the above information allows us to obtain (3.23) and (3.24) at time k. It then follows that $C_k^{\text{low}} \leq \mathsf{C} \leq C_k^{\text{up}}$. By induction, we then obtain (3.23) and (3.24) for $k = 1, \cdots, K-1$.

3.4.4 Problem Statement

We are now ready to state our throughput-maximization problem. Consider a frame that consists of packets $1, 2, \cdots, K$. Given CSI C_k at time k, we wish to optimally choose a policy $\mathcal{P}$ so as to maximize the throughput summed over the entire frame. That is, our problem $\mathsf{P}_{\max}$ is given by

$$\mathsf{P}_{\max}: \qquad \max_{\mathcal{P}} \mathbb{E}\left[\sum_{k=1}^{K} T_k(\mathcal{P}; C_k)\right]$$

subject to constraints (3.16), (3.17), (3.18) for all k, i. The expectation is taken over the mutual information C (since h is a random variable) and consequently over the policy and CSI. We denote the optimal policy as $\mathcal{P}^*$ and the maximum sum throughput as T_{sum}^*.

3.4.5 Optimal Policy by Dynamic Programming

The optimal policy can be obtained, in principle, by dynamic programming. Let $J_k(C_k)$ be the maximum expected throughput sum from packet k to packet K by varying the policy, given CSI C_k at packet k. From this definition, we have $T_{\mathsf{sum}}^* = J_1(C_1)$. We recall that $C_1 = \emptyset$ since no CSI is available initially, hence there is only one possible initial CSI state. By dynamic programming [65], J_1 can be obtained by a recursive computation with decreasing k based on the Bellman's equations (3.25), (3.26) given below. For $k = K$, we have for all possible C_K

$$J_K(C_K) = \max_{\mathcal{P}_K(C_K)} T_k(\mathcal{P}; C_K), \tag{3.25}$$

while for $k = K-1, \cdots, 1$, we have for all possible C_k

$$J_k(C_k) = \max_{\mathcal{P}_k(C_k)} \{T_k(\mathcal{P}; C_k) + \mathbb{E}_{C_{k+1}|C_k}\left[J_{k+1}(C_{k+1})\right]\}. \tag{3.26}$$

Specifically, we begin the recursion by first computing (3.25) for all possible C_K. We then use previously computed results to obtain (3.26) for all C_k, where k is decreased by one for each recursion according to $k = K-1, \cdots, 1$. From results in dynamic programming [65], the optimal policy $\mathcal{P}^*$ is given by the set of optimal parameters $\mathcal{P}_k^*(C_k)$ that maximizes (3.25), (3.26) for all C_k and for all k.

3.5 Equal-Rate Condition (ERC)

Although an optimal policy $\mathcal{P}^*$ can be obtained by dynamic programming, the complexity of computation is prohibitive when K becomes large. This is because the number of possible $\mathbf{a}_K$ increases exponentially according to 2^K, since each ACK can be a PACK or an NACK. Moreover, for a given $\mathbf{a}_k$, the number of parameters $\mathcal{P}_k$ can grow linearly with k. For instance, in the worst case when k NACKs are received, there are $k-1$ possible $\alpha_{k,i}$'s that need to be optimized.

To reduce the computational complexity, we seek to impose conditions to limit the number of parameters. Ideally, the resulting loss in the optimal throughput should be small or negligible. To this end, we observe that the throughput T_k in (3.22) depends largely on a *single* rate $R_k^{\max}$. Thus, it is reasonable to encode all messages sent out so far at the *common* effective rate R_k^{eq}; effectively, we have a single super-packet encoded at the common rate of $R_k^{\max} = R_k^{\text{eq}}$. Consequently, the ACK bit is used to reflect whether this super-packet is received correctly, i.e., whether $R_k^{\text{eq}} < \mathsf{C}$. This motivates us to impose the ERC on the rate-adaptation policy, as follows.

Definition: The set of parameters for packet k in $\mathcal{P}_k$ is said to satisfy the ERC if the effective rates of all yet-to-recover messages $\{w_i, i \in \mathcal{S}_k\}$ in packet k take a common value R_k^{eq}, i.e.,

$$R_{k,i} = R_k^{\text{eq}},\ i \in \mathcal{S}_k. \tag{3.27}$$

This implies that $R_k^{\max} = R_k^{\text{eq}}$ according to the definition in (3.14).

Definition: A policy $\mathcal{P}$ is said to satisfy the ERC if its parameters $\mathcal{P}_k$ for packet k satisfy the ERC for $k = 1, \cdots, K$.

In our problem, we have restricted the feedback to a single ACK bit per packet. That is, the receiver can at most inform the transmitter whether the channel can support a particular transmission rate. If the transmitter chooses a rate for the incremental data that is different from the rate of a yet-to-recover message, then the receiver cannot completely inform the transmitter which of the messages have been recovered. Hence, we conjecture that it is sufficient to send all coded bits at the same effective rate so as to optimize the use of the feedback ACK, i.e., there is no loss in the maximum achievable throughput if we only consider policies that satisfy the ERC. However, we have not been able to prove this conjecture[6].

[6]To prove the conjecture, we have to show that the ERC is optimal for problem $\mathsf{P}_{\max}$. It is sufficient to show that (i) the ERC is optimal in maximizing the current throughput and (ii) the ERC does not lead to a loss to the throughput in the future. Although condition (i) can be shown to hold, we have not been able to prove that condition (ii) holds.

3.5.1 Simplifications

Let $m \geq 0$ denote the length of the NACKs, i.e., $m = k - \tilde{k}$. We note that $\alpha_{k,i}$ can be obtained using (3.10) and (3.27) as

$$\alpha_{k,i} = \beta_i / R_k^{\text{eq}} - \sum_{j=i}^{k-1} \alpha_{j,i}, \quad i \in \mathcal{S}_{k-1}. \tag{3.28}$$

We will show generally that the ERC leads to a significant reduction in $\mathcal{P}$ and the space of C_k, thus simplifying the computation of Bellman's equation. We separately consider the cases of $m > 0$ and $m = 0$.

(i) $m > 0$: Suppose that $k > \tilde{k}$, or equivalently $m > 0$. This corresponds to the case of a NACK being received in the previous packet. Substituting (3.28) into the constraint (3.18) we can then obtain

$$R_k^{\text{eq}} = \mathcal{B}_k/(m+1). \tag{3.29}$$

Since $\mathcal{B}_k = \mathcal{B}_{k-1} + \beta_k$, we can alternatively write

$$R_k^{\text{eq}} = m/(m+1) R_{k-1}^{\text{eq}} + \beta_k/(m+1) \tag{3.30}$$

where R_{k-1}^{eq} is defined similarly as in (3.29) with k replaced by $k-1$. Since $\beta_k \geq 0$, it follows that $R_k^{\text{eq}} \geq m/(m+1) R_{k-1}^{\text{eq}}$. Moreover, since a NACK is received for the previous packet, to optimize throughput R_k^{eq} must not be increased with respect to R_{k-1}^{eq}. Thus, R_k^{eq} must be chosen between

$$m/(m+1) R_{k-1}^{\text{eq}} \leq R_k^{\text{eq}} \leq R_{k-1}^{\text{eq}}. \tag{3.31}$$

Finally, the throughput (3.22) can now be simplified using (3.29) as

$$T_k(R_k^{\text{eq}}; \widetilde{C}_k) = (m+1) R_k^{\text{eq}} \Pr(R_k^{\text{eq}} \leq \mathsf{C} | C_{k-1}^{\text{low}} \leq \mathsf{C} \leq C_{k-1}^{\text{up}}). \tag{3.32}$$

subject to (3.31). The rate R_{k-1}^{eq} can be considered to be part of the CSI that is used for rate adaptation.

(ii) $m = 0$: Suppose that $k = \tilde{k}$, or equivalently $m = 0$. This corresponds to the case of a PACK being received in the previous packet. Clearly, (3.29), (3.30) and (3.32) apply without loss of generality. However, (3.31) does not apply. In fact, since a PACK is received previously and so the channel can support at least the rate R_{k-1}^{eq}, to maximize throughput we have $R_k^{\text{eq}} \geq R_{k-1}^{\text{eq}}$.

In general, for $m \geq 0$, we observe from (3.32) that the only parameter that needs to be adapted now is the ERC rate R_k^{eq}. Thus, the set of parameters $\mathcal{P}_k$ in (3.15) can now be replaced by

$$\widetilde{\mathcal{P}}_k(\mathbf{a}_{k-1}) = \{R_k^{\text{eq}}(\mathbf{a}_{k-1})\}, \tag{3.33}$$

which contains only *one* rate parameter given $\mathbf{a}_{k-1}$, regardless of the actual value of $\mathbf{a}_{k-1}$.

Moreover, the original CSI C_k in (3.19) can be reduced. From (3.31) and (3.32), it is clearly sufficient to use the *reduced CSI*

$$\widetilde{C}_k = \{m, R_{k-1}^{\text{eq}}, C_{k-1}^{\text{low}}, C_{k-1}^{\text{up}}\}. \tag{3.34}$$

The dimension of this CSI space is four[7].

Finally, we can reduce the space of the reduced CSI, by noting that R_{k-1}^{eq} in $\widetilde{C}_k$ can be bounded for $k \geq 2$ as

$$C_{k-1}^{\text{low}} \leq R_{k-1}^{\text{eq}} \leq C_{k-1}^{\text{up}}. \tag{3.35}$$

To see this, we note that to maximize throughput, R_{k-1}^{eq} has to be chosen such that

$$C_{k-2}^{\text{low}} \leq R_{k-1}^{\text{eq}} \leq C_{k-2}^{\text{up}}, \tag{3.36}$$

since it is known (from previous CSI) that C must lie between C_{k-2}^{low} and C_{k-2}^{up}. If packet $k-1$ is received correctly, from (3.23) and (3.24) we get $C_{k-1}^{\text{low}} = C_{k-2}^{\text{low}}$ and $C_{k-1}^{\text{up}} = R_{k-1}^{\text{eq}}$, respectively. Substituting both results into (3.36), we then obtain $C_{k-1}^{\text{low}} \leq R_{k-1}^{\text{eq}} = C_{k-1}^{\text{up}}$, and thus (3.35) holds. Otherwise, if packet $k-1$ is not received correctly, from (3.23) and (3.24) we get $C_{k-1}^{\text{low}} = R_{k-1}^{\text{eq}}$ and $C_{k-1}^{\text{up}} = C_{k-2}^{\text{up}}$, respectively. Substituting both results into (3.36), we then obtain $C_{k-1}^{\text{low}} = R_{k-1}^{\text{eq}} \leq C_{k-1}^{\text{up}}$, and thus (3.35) holds. Hence, (3.35) holds in general.

3.5.2 A Graphical Interpretation

The ERC also offers a simple interpretation of the adaptation process, which is shown graphically in Fig. 3.4. The discrete time index is reflected on the x-axis, while the accumulated throughput that is yet to be recovered is reflected on the y-axis. From (3.29), the ERC rate used for packet k is given by $R_k^{\text{eq}} = \mathcal{B}_k/k$, which can be interpreted as the gradient of the line from the origin $(0,0)$ to $(k, \mathcal{B}_k)$. In Fig. 3.4, we also draw a line with the capacity C as its gradient. By comparing these gradients, we thus see immediately that $R_3^{\text{eq}} < \mathsf{C} < R_2^{\text{eq}} < R_1^{\text{eq}}$. Hence, we would receive NACKs for packets $1, 2$ and a PACK for packet 3.

During rate adaptation, C is not known exactly, but we may initially have some priori knowledge of the minimum and maximum possible capacity $C^{\text{low}}, C^{\text{up}}$, for instance, according to (3.35) (not shown graphically here). These would be reflected as gradients that should bound R_k^{eq}. Once a PACK is obtained, the rate adaptation process resets to start at the origin, with an updated $C^{\text{low}}, C^{\text{up}}$.

[7] We can further reduce the dimension by one, without loss in optimality in maximizing throughput, as follows. If packet $k-1$ is received correctly, from (3.24) we have $R_{k-1}^{\text{eq}} = C_{k-1}^{\text{up}}$; otherwise if packet $k-1$ is not received correctly, from (3.23) we have $R_{k-1}^{\text{eq}} = C_{k-1}^{\text{low}}$. Essentially, we need to consider only a three-dimensional CSI space.

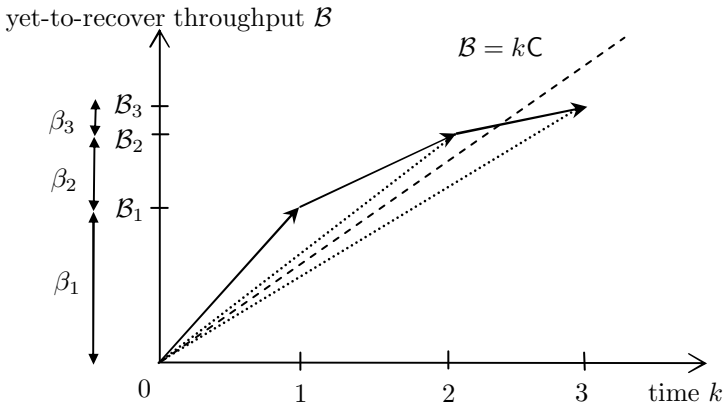


Fig. 3.4: A graphical interpretation of a typical rate adaptation with ERC. The ERC rate used for packet k is given by $R_k^{\text{eq}} = \mathcal{B}_k/k$, where $\mathcal{B}_k = \sum_{i=1}^{k} \beta_i$ and β_i is the amount of incremental bits sent in packet i.

As an illustration of the use of this graphical interpretation, we note that given (3.36), Fig. 3.4 allows (3.35) to be easily deduced and understood in a graphical manner. Graphically speaking, to maintain optimality, we should perform rate adaptation for packet k such that the gradient of the line connecting $(0, 0)$ to $(k, \mathcal{B}_k)$ lies between the gradients of C_{k-1}^{low} and C_{k-1}^{up}.

3.6 Examples and Numerical Results

3.6.1 Proposed Coding Scheme with ERC

We obtain numerical results for the maximum throughput achieved by our proposed coding scheme with ERC. To obtain our numerical results, we concatenate all ERC rates $R_k^{\text{eq}}(\mathbf{a}_k)$, for all possible $\mathbf{a}_k$ and for $k = 1, \cdots, K$. This gives a rate vector $\mathbf{r}_{\mathsf{all}}$ of length $2^K - 1$ with elements that are constrained by (3.31). The problem $\mathsf{P}_{\max}$ under ERC becomes

$$\widetilde{\mathsf{P}}_{\max}: \qquad \max_{\mathbf{r}_{\mathsf{all}}} \widetilde{T}_{\mathsf{sum}}$$

where $\widetilde{T}_{\mathsf{sum}} \triangleq \mathbb{E}_{\mathbf{a}_K}\left[\sum_{k=1}^{K} T_k(R_k^{\text{eq}}(\mathbf{a}_{k-1}); \widetilde{C}_k)\right]$.

For clarity, we denote the maximum throughput when the frame consists of K packets as $\widetilde{T}_{\mathsf{sum}}^*(K)$. For example, consider $K = 2$. The rate vector becomes $\mathbf{r}_{\mathsf{all}} = [R_1^{\text{eq}}, R_2^{\text{eq}}(0), R_2^{\text{eq}}(1)]$. After some simplifications, the accumulated throughput $\widetilde{T}_{\mathsf{sum}}(2)$

becomes

$$\begin{aligned} & R_1^{\mathrm{eq}} \times \Pr(R_1^{\mathrm{eq}} \leq \mathsf{C}) \\ +& R_2^{\mathrm{eq}}(1) \times \Pr(R_1^{\mathrm{eq}} \leq \mathsf{C}, R_2^{\mathrm{eq}}(1) \leq \mathsf{C}) \\ +& 2R_2^{\mathrm{eq}}(0) \times \Pr(R_1^{\mathrm{eq}} > \mathsf{C}, R_2^{\mathrm{eq}}(0) \leq \mathsf{C}) \end{aligned} \tag{3.37}$$

subject to the constraint

$$R_1^{\mathrm{eq}}/2 \leq R_2^{\mathrm{eq}}(0) \leq R_1^{\mathrm{eq}}, \tag{3.38}$$

according to (3.31). In (3.37), the first term is the throughput of packet 1. The second term is the throughput of packet 2 when packet 1 is received correctly, i.e., when $R_1^{\mathrm{eq}} \leq \mathsf{C}$. The third term is the throughput of packet 2 when packet 1 is *not* received correctly, i.e., when $R_1^{\mathrm{eq}} > \mathsf{C}$. The accumulated throughput for $K > 2$ can be obtained likewise. Then, we numerically optimize $\mathbf{r}_{\mathsf{all}}$ by using the constrained optimization function `fmincon.m` that is available in the MATLAB software.

We obtain the throughput achieved using IRIDC, assuming a Rayleigh fading channel at different SNR. We plot $\widetilde{T}^*_{\mathsf{sum}}(K)/K$ for $K = 1, 2, 3, 4$ in Fig. 3.5. As an upper bound, we consider the ergodic capacity, given by $\mathbb{E}[\mathsf{C}]$. This bound is achieved when the transmitter has full CSI of the channel, by matching the rate to the capacity C for every transmission. On the other hand, a lower bound that acts as a benchmark is given by $\widetilde{T}^*_{\mathsf{sum}}(1)$. This corresponds to the case when the memory of the system is reset after every packet. Hence, neither IR nor ID is sent, and also rate adaptation cannot be performed. From Fig. 3.5, we see that as K increases, the maximum throughput improves significantly, especially when K is small. In particular, we see that $\widetilde{T}^*_{\mathsf{sum}}(4)/4$ is only less than one bit of throughput capacity from the ergodic capacity when the SNR is less than 40 dB. Hence, a frame size of $K = 4$ packets is sufficient for most practical scenarios.

3.6.2 Comparison with Known Schemes

For comparison, we consider throughput maximization using the IC and IRC schemes. In both schemes, each ACK A_k is used as CSI that informs the transmitter whether the capacity C is above or below some rate R_k. For illustration, we consider $K = 2$.

3.6.2.1 IC

For IC, there exists Gaussian codes that satisfy [71]

$$R_k < \mathsf{C} \Leftrightarrow w_k \text{ is decoded successfully at time } k \tag{3.39}$$

with high probability for large N. Assuming (3.39) to hold exactly, the accumulated throughput using IC simplifies as

$$\begin{aligned} & R_1 \times \Pr(R_1 \leq \mathsf{C}) \\ +& R_2(1) \times \Pr(R_1 \leq \mathsf{C}, R_2(1) \leq \mathsf{C}) \\ +& R_2(0) \times \Pr(R_1 > \mathsf{C}, R_2(0) \leq \mathsf{C}). \end{aligned} \tag{3.40}$$

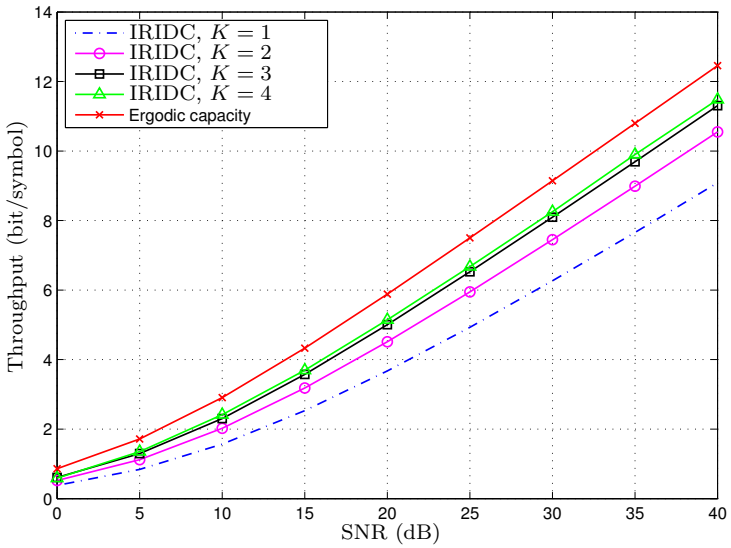


Fig. 3.5: Maximum throughput achieved using IRIDC scheme with ERC. Rate adaptation is carried over a frame of $K = 1, 2, 3$ or 4 packets.

To maximize this throughput, rate $R_2(0)$ has to be chosen to satisfy

$$R_2(0) \leq R_1. \tag{3.41}$$

This is because if $R_2(0) > R_1$, the last line in (3.40) becomes zero and so $R_2(0)$ cannot be optimal.

We now compare the accumulated throughput of IRIDC and IC, subject to their corresponding rate constraints. For comparison purposes, we may treat all rates as variables, subject to certain constraints, that can be optimized so as to maximize the accumulated throughput. We see that the constraint (3.38) is more restrictive than (3.41). If we ignore both constraints, the difference of both accumulated throughput (3.37), (3.37) lies in the last line, when packet 1 is not received correctly. Clearly, (3.37) can then always achieve a larger throughput than (3.40) since the throughput that is achieved in the last line is doubled. Numerical results show that the constraint (3.38) is typically not overly restrictive compared to (3.41), but the improvement in throughput due to the doubling of the throughput in the last line of (3.38) is much more substantial.

3.6.2.2 IRC

For IRC, we recall that $\tilde{k}$ denotes the packet index of the first packet in the recent burst of NACKs that last until packet k. There exists Gaussian codes that satisfy [70]

$$\frac{R_{\tilde{k}}}{i - \tilde{k} + 1} < \mathsf{C} \Leftrightarrow w_{\tilde{k}} \text{ is decoded successfully at time } i \tag{3.42}$$

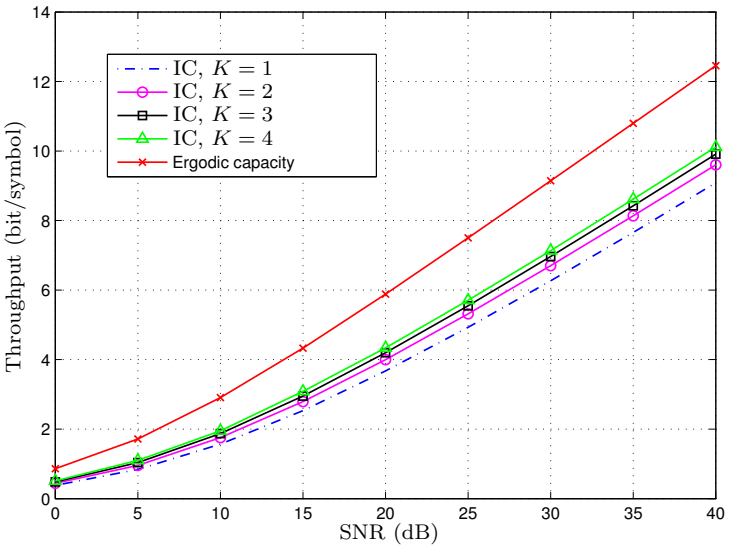


Fig. 3.6: Maximum throughput achieved using IC scheme. Rate adaptation is carried over a frame of $K = 1, 2, 3$ or 4 packets.

for $i \in \mathcal{S}_k$, with high probability for large N. Assuming (3.42) to hold exactly, for $K = 2$ the accumulated throughput simplifies as

$$\begin{aligned} & R_1 \times \Pr\left(R_1 \leq \mathsf{C}\right) \\ + & R_2(1) \times \Pr\left(R_1 \leq \mathsf{C}, R_2(1) \leq \mathsf{C}\right) \\ + & R_1 \times \Pr\left(R_1 > \mathsf{C}, R_1/2 \leq \mathsf{C}\right). \end{aligned} \tag{3.43}$$

We see that (3.38) reduces to (3.43) if we select $R_2^{\text{eq}}(0)$ as the lower bound of (3.38). Hence, the maximum throughput achieved with IRIDC is always equal or greater than IRC.

3.6.2.3 Numerical Results

Similar to the previous section, we perform simulations based on Rayleigh fading channels. We obtain the maximum throughput achieved using IC scheme in Fig. 3.6 and using IRC scheme in Fig. 3.7. We vary $K = 2, 3, 4$. In both figures, we see that the maximized throughput lies between the benchmark given by $\widetilde{T}^*_{\mathsf{sum}}(1)$ and the upper bounds given by the ergodic capacity. Moreover, the throughput increase with K. However, we see that the IC scheme generally performs worse than the IRC scheme. This is because the IRC scheme is more robust, since even if a packet is not received correctly, the message to be delivered by the erroneous packet can still be recovered later.

Finally, we compare the performance of the IRIDC scheme with the IC and IRC schemes, for $K = 4$. The maximized throughput is shown in Fig. 3.8. We see that

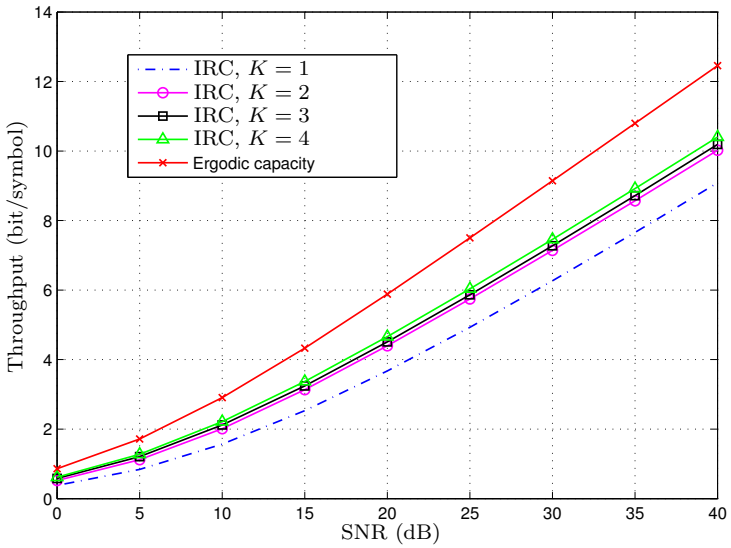


Fig. 3.7: Maximum throughput using IRC scheme. Rate adaptation is carried over a frame of $K = 1, 2, 3$ or 4 packets.

the IRIDC scheme can clearly achieve a larger throughput than the other schemes. This observation holds also for $K = 2, 3$, but at a smaller degree.

3.7 Packet Delay and Packet Outage

We have specifically optimized rate adaptation with the proposed IRIDC scheme to maximize the throughput. Hence, the optimal rate adaptation policy does not attempt to minimize the packet delay or the packet outage. In practice, the packet delay or packet outage may be more important design parameters for some applications. We give a qualitative discussion on these two issues when the IRIDC scheme is employed, compared to the IC and IRC schemes.

To formulate our problem, we have imposed a block fading model, in which the system memory is reset after every K packets are transmitted. This scenario in fact imposes a hard deadline on the delay, since any data bit that is not successfully transmitted is equivalently treated as lost and discarded. Hence, although we do not specifically consider packet delays in our problem, with the IRIDC scheme we may treat the parameter K as a delay constraint (provided that the channel remains invariant over K packets, as assumed in our model). This delay constraint also applies when the IRC scheme is employed, but may not applies for the IC scheme because every packet in the IC scheme is always dropped after it is transmitted.

As a side result of the numerical study conducted in the previous section, we have observed that by using the IRIDC scheme with optimized rate adaptation, the packets

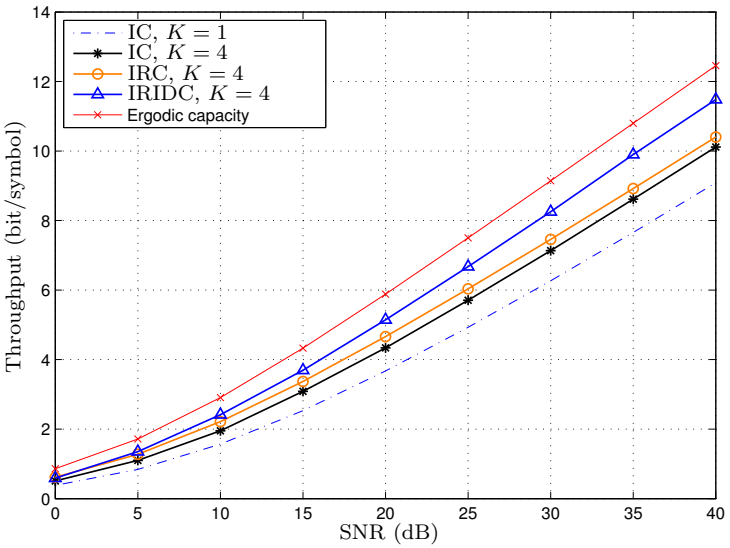


Fig. 3.8: Comparison of maximum throughput achieved using IC, IRC and IRIDC schemes. Rate adaptation is carried over a frame of $K = 4$ packets. Note that for $K = 1$, the IC, IRC and IRIDC schemes are equivalent.

typically suffer from a high packet outage probability. This high-outage phenomenon, in the range of $10\% - 30\%$ outage probability, occurs especially for the initial transmissions. This is intuitively reasonable since in the IRIDC scheme, redundancy are sent after a failed transmission. In order to maximize throughput, the optimized rate adaptation policy attempts to transmit at a high rate (resulting in a high outage probability initially), failing which it sends some redundancy to recover the failed packet. This high-outage phenomenon applies for the IRC scheme but less so for the IC scheme. This is because the IRC scheme, but not the IC scheme, sends redundancy subsequently after a failed transmission.

Nevertheless, since our objective focuses on throughput maximization, for any of the schemes considered in this chapter, the resulting policy might not be appropriate for delay-sensitive or outage-sensitive applications. In such cases, the delay and outage probability has to be considered explicitly in the problem formulation, either in the objective function by penalizing large delay or high outage probability, or explicitly as hard constraints in the optimization problem.

3.8 Conclusion

We propose the incremental-redundancy incremental-data coding (IRID) scheme for ARQ. This scheme allow redundancy to be sent to help to recover previous erroneous packets, and also new information bits to be sent to optimize the use of the channel

resources. To illustrate the potential of the IRIDC scheme, we consider the maximization of the throughout by rate adaptation in a block-fading channel. We assume the availability of lean channel state information at the transmitter, in the form of acknowledgement (ACK) bits. To simplify rate adaptation, we encode the new information bits at the same rate as the effective rate of the erroneous information bits. Numerical results obtained for Rayleigh fading channels show that substantial improvement in throughput can be achieved using the IRIDC scheme, compared to conventional ARQ schemes.

CHAPTER 4

ITERATIVE SUBCARRIER RECONSTRUCTION IN OFDM SYSTEMS

In this chapter[1], we consider an uncoded pre-transformed (PT) orthogonal frequency division multiplexing (OFDM) system where the channel is not known at the transmitter. We propose an iterative detector that performs well, yet is low in complexity. To keep the complexity low, a linear filter is used for detection in each iteration. To reduce the noise enhancement arising from the linear filter, we introduce a reconstruction step. Specifically, in the ith iteration, the reconstruction step replaces the received signal at the ith worst subcarrier (i.e., with the ith smallest channel amplitude) with an estimate based on previous data symbol decisions. The transform used in the PT-OFDM and the reconstruction step are designed to maximize the worst SNR across all subcarriers. Under the assumption that previous decisions are correct, we show analytically that the iterative detector achieves a diversity advantage of $i+1$ in the ith iteration, thus explaining its superior performance. Due to the flexibility of the transform design, the analysis conducted is applicable for many common PT-OFDM systems. Simulations for realistic channels show a superior bit error rate performance of the iterative detector compared to conventional detectors.

[1] A large part of this work has been published as "Iterative detection for pretransformed OFDM by subcarrier reconstruction" in *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2842–2854, Aug. 2005.

4.1 Introduction

Future wireless systems have to achieve high spectral efficiency to enjoy high user capacities and high data rates. Multicarrier modulation realized by orthogonal frequency division multiplexing (OFDM) [31] [72] [73] is well suited for high data rate applications in fading channels and has been chosen for several broadband wireless LAN standards: IEEE 802.11a, European HIPERLAN/2, and Japanese multimedia mobile access communication systems [74].

An OFDM system uses a cyclic prefix (CP) to remove inter-block interference. The CP also transforms a frequency-selective fading channel into multiple flat fading parallel transmission *subcarriers* in the frequency domain. In such a system, the information sent on some carriers might be subject to strong attenuation and might not be recovered correctly at the receiver. This has motivated the proposal of more robust transmission schemes combining the advantages of code division multiple access (CDMA) [75] with the advantages of OFDM, in which the information is spread across all the subcarriers by some pre-transformation (PT) matrix, referred to here in general as PT-OFDM. For specific transforms, PT-OFDM is also known as OFDM-CDMA [76], spread OFDM [77], WHT-OFDM [78,79], or multi-carrier (MC)-CDMA [73]. PT-OFDM has been investigated in the European Information Society Technologies program, as a promising candidate for future wireless communications with a very high data rate transmission [78]. The single carrier frequency domain equalization (SC-FDE) [80] [81] system is also a special case of a PT-OFDM system, where the pre-transformation is equal to the Fourier transform. It has been proposed to the IEEE 802.16 working group as a possible solution for a broadband wireless metropolitan area networks [81].

The PT-OFDM system can be classified into a broad class of linear block-based transmission schemes [82]. A general treatment of the linear block-based system is given in [82, 83], where the inter-block interference can be removed via the use of zero padding or CP. System designs optimized with respect to signal-to-noise ratio (SNR) or bit error rate (BER) are derived in [82] and [83], respectively. However, knowledge of the channel is required at the transmitter for these schemes.

We focus on the case when CP is used, which is by far the most common choice considered in the literature. System designs without assuming channel knowledge at the transmitter have also been carried out in the literature, such as in [34, 35, 84]. In [34], the transform used for the PT-OFDM system is optimized so as to minimize the BER when a linear least squares (LS) or minimum mean squared error (MMSE) detector is used. The BER can be improved by exploiting further possible diversity advantage, in particular by not restricting to the use of linear detectors. In [35, 84], the diversity advantage is fully exploited by a maximum likelihood (ML) detector. However, high detection complexity is required, which increases exponentially with the size of the transform. Although it is demonstrated in [84] that superior performance can be obtained if linear detectors are used for detecting BPSK modulated symbols using the same transform design, the BPSK performance is not as promising as when complex-valued symbols are used. In [37, 38], an iterative detector based on parallel

interference cancellation (PIC) is proposed for multi-user detection. In this context of PT-OFDM, the number of users and transform size are equal and are usually large (say 64). The PIC when applied for the PT-OFDM system will therefore result in a high complexity detector.

When the ML detector is used, the choice of transform affects the system performance substantially. A transform optimized for the ML detector is proposed in [36]. Another related work [85] designs the transforms to be used in a novel manner. A BPSK data stream is split into two data streams which are transmitted concurrently via different transforms in an OFDM system (akin to different codes in a multiuser case). The orthogonal transforms are designed to minimize the inter-code interference. However, when complex symbols are used, obtaining a superior performance is not straightforward. In [86], the asymptotic performance of the PT-OFDM system with a randomly chosen transform is investigated, as the transform size goes to infinity.

The BER is usually the most important performance measure. However, the transform in PT-OFDM can be designed to cater specifically for other attractive properties as well, such as lower clipping probability, less spectral re-growth, and lower block error rate as explored in [78, 87, 88]. These could be used as secondary optimizing objectives when a sufficient degree of freedom is available in the transform design.

The main contributions of this chapter are as follows.

- We introduce a novel low complexity iterative detector that improves the BER performance at a very low computational cost. The iterative detection starts with a linear LS or MMSE detector. The received signal at a certain subcarrier is replaced by an estimated signal based on information of all the previously detected symbols. We called this process *reconstruction*. The process of data detection and reconstruction is carried out iteratively.
- We optimize the transform coefficients and reconstruction method so as to maximize the minimum SNR. A class of unitary matrices with constant magnitude satisfies the transform design requirements. This large degree of freedom allows one to choose the transform to satisfy some secondary objectives as well. In addition, reconstruction should be carried out starting from the subcarrier with the smallest channel amplitude (we called this the *weakest* subcarrier). On the next iteration, the next weakest subcarrier is selected for reconstruction, and so on.
- We analyze the performance of the proposed iterative detector by bounding the SNR appropriately, under the ideal assumption that the previous detections are error free. Our main result is that, for i.i.d. Rayleigh-distributed subcarrier channels, the iterative method achieves a diversity advantage of $i + 1$ after the ith iteration. The simulation results, for realistic channels, illustrate that the BER performance of the iterative detection is much better than that of the conventional detectors for PT-OFDM or OFDM system.

The proposed iterative detector differs from the algorithms commonly used in the multiuser detection context, such as PIC. The conventional iterative process proceeds user by user (corresponding to OFDM symbol by symbol here) to cancel multiuser/intersymbol interference. Our method involves a reconstruction and detection

process instead, subcarrier by subcarrier. In principle, both PIC and the proposed iterative detector can be implemented together to further improve the system performance, but at a larger cost in complexity. We emphasis that this subcarrier-by-subcarrier based iterative detection method does not apply to conventional OFDM systems, since data symbols in a conventional OFDM system are transmitted independently on orthogonal subcarriers. It is the presence of the pre-transformation that spreads each of the data symbols on all subcarriers that permits the success of the iterative method.

This chapter is organized as follows. First, the PT-OFDM system model is described in Section 4.2. The iterative detection method is given in Section 4.3, whereby the transform coefficients and reconstruction method are derived to maximize the minimum SNR across different symbols. The flexibility of the transform design is explored and refinements of the algorithm are suggested in order to realize better performance. In Section 4.4, assuming perfect reconstruction, we obtain closed-form bounds on the BER of the iterative detector by using order statistics. A comparison of the analytical performance of the PT-OFDM system with and without iterations is also presented. Finally simulations are conducted in Section 4.6 to illustrate the performance of the proposed detector and conclusions are given in Section 4.7.

Throughout this chapter the following notations are adhered to. Bold lower case letters are used to denote column vectors. Bold upper case letters are used to denote matrices. The superscripts *, ^{T} and ^{H} are used to denote complex conjugate, transpose and Hermitian, respectively. The (m,n)th element of matrix $\mathbf{W}$ and mth element of vector $\mathbf{w}$ is denoted as w_{mn} and w_m, respectively. The identity matrix is denoted as $\mathbf{I}$.

4.2 System Description

4.2.1 PT-OFDM

The PT-OFDM system block diagram is depicted in Fig. 5.1. For simplicity, we consider a system with $M = 2^k$ subcarriers where M information symbols $x_m, m = 1, 2, \cdots, M$, are transmitted at the same time in one OFDM symbol. The modulation symbols s_m of the subcarriers are calculated from the information symbols x_m using the matrix operation

$$\mathbf{s} = \mathbf{W}\mathbf{x} \tag{4.1}$$

where $\mathbf{s} = [s_1, s_2, \ldots, s_M]^T$, $\mathbf{x} = [x_1, x_2, \ldots, x_M]^T$ and $\mathbf{W}$ represents the PT matrix of size $M \times M$. Therefore, there is no loss of code rate in terms of the number of information symbols transmitted per channel use. In the case of an OFDM system, $\mathbf{W}$ is simply an identity matrix. The block of modulation symbols $\mathbf{s}$ is then passed through an inverse discrete Fourier transform, which can be implemented efficiently using the inverse fast Fourier transform (IFFT). After inserting a cyclic prefix with duration no shorter than the maximum channel delay spread, the PT-OFDM symbol is

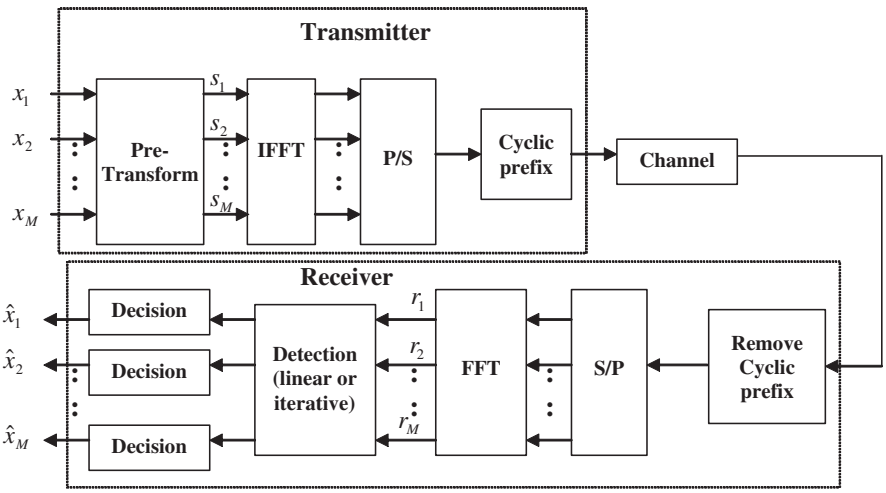


Fig. 4.1: PT-OFDM system block diagram. The detection block at the receiver can be implemented either with a linear filter or with an iterative procedure involving linear filters.

transmitted. The channel is assumed to be a quasi-static frequency-selective Rayleigh fading channel corrupted by additive white Gaussian noise (AWGN).

At the receiver, the samples of the received signal corresponding to the cyclic prefix are removed. The output vector after FFT, $\mathbf{r} = [r_1, r_2, \ldots, r_M]^T$, can be written as

$$\mathbf{r} = \mathbf{\Gamma} \cdot \mathbf{s} + \mathbf{n} = \mathbf{\Gamma} \cdot \mathbf{W} \cdot \mathbf{x} + \mathbf{n} \tag{4.2}$$

where $\mathbf{\Gamma} = \text{diag}(h_1, h_2, \ldots, h_M)$, a diagonal matrix and $\mathbf{n}$ is the $M \times 1$ AWGN vector with independent elements each of which is zero mean and has variance σ_n^2. The elements $h_1, h_2, \ldots, h_M$ are the frequency-domain channel coefficients, given as

$$h_m = \sum_n \tilde{h}_n \exp(-j2\pi nm/M), m = 1, 2, \cdots, M.$$

Here, we assume a sample-spaced L_hth order FIR channel model with time-domain channel taps $\tilde{h}_n, n = 0, \cdots, L_h$.

4.3 Detection Algorithms

The iterative detection algorithm consists of three stages for each iteration, namely reconstruction, linear filtering, and decision. For clarity, we shall describe the simplest method to implement each stage, and discuss the possible variations of each stage at the end of the section.

4.3.1 Initialization

The initial iteration skips the step of reconstruction but proceeds by suppressing the interference amongst different information symbols. We minimize the squared error after linear filtering by using the LS criterion. Thus, the received signal vector $\mathbf{r}$ has to be linearly filtered using

$$\mathbf{G} = \mathbf{W}^{-1} \cdot \mathbf{\Gamma}^{-1} = \mathbf{W}^{-1} \cdot \mathrm{diag}(1/h_1, 1/h_2, \ldots, 1/h_M). \tag{4.3}$$

This is also known as the zero-forcing (ZF) or orthogonality restoring combining (ORC) detector in the MC-CDMA literature in a synchronous system [73, 89]. As weak subcarriers are amplified with high gain, they introduce a high noise level.

Applying the LS filter $\mathbf{G}$ on the received signal $\mathbf{r}$ from (5.1), we get

$$\widetilde{\mathbf{x}}_0 \triangleq \mathbf{G} \cdot \mathbf{r} = \mathbf{x} + \mathbf{W}^{-1}\mathbf{\Gamma}^{-1}\mathbf{n}. \tag{4.4}$$

The jth element of $\widetilde{\mathbf{x}}_0$, denoted as $\widetilde{x}_{0,j}$, can be written as

$$\widetilde{x}_{0,j} = x_j + \sum_{m=1}^{M} \frac{\psi_{jm} n_m}{h_m}, \quad j = 1, 2, \cdots, M \tag{4.5}$$

where ψ_{jm} is the (j,m)th element of the matrix $\mathbf{\Psi} \triangleq \mathbf{W}^{-1}$.

We then perform a hard decision on the filtered signal to get

$$\hat{\mathbf{x}}_0 = \mathrm{dec}(\widetilde{\mathbf{x}}_0),$$

where $\mathrm{dec}(\cdot)$ stands for the decision function. Denote $\hat{\mathbf{x}}_0 = \mathbf{x} + \mathbf{e}_0$ where $\mathbf{e}_0$ represents the error vector due to wrong decision made. These procedures constitute the initial iteration (we refer to this as iteration $i = 0$). If required, $\hat{\mathbf{x}}_0$ can be used to detect the transmitted signal. However, further iterations can result in better performance.

4.3.2 Subsequent Iterations

First, we attempt to estimate the noiseless received signal vector $\hat{\mathbf{x}}_0$ as follows

$$\widetilde{\mathbf{r}}_1 \triangleq \mathbf{\Gamma}\mathbf{W}\hat{\mathbf{x}}_0 = \mathbf{r}_{noiseless} + \widetilde{\mathbf{e}}_1 \tag{4.6}$$

where $\mathbf{r}_{noiseless} = \mathbf{\Gamma}\mathbf{W}\mathbf{x}$ represents the received signal when noise is absent, and $\widetilde{\mathbf{e}}_1 = \mathbf{\Gamma}\mathbf{W}\mathbf{e}_0$ represents the introduced error due to the decision error $\mathbf{e}_0$. From this, we see that the reconstructed vector $\widetilde{\mathbf{r}}_1$ removes the noise from the receiver but introduces error due to the wrong detection in the previous iteration. In other words, we trade noise $\mathbf{n}$ with error $\mathbf{n}$. This trade-off is not worthwhile on average since the likelihood of a wrong symbol detection is high. Thus, it is probable that this will result in an even larger increase in interference due to error propagation rather than only the anticipated reduction of noise.

Let us denote $\boldsymbol{\mathcal{I}}_m$ as a diagonal matrix with value 1 on its mth diagonal term and 0 otherwise, and $\boldsymbol{\mathcal{E}}_m$ as a diagonal matrix with value 0 on its mth diagonal term and 1 otherwise[2]. Consider only substituting one element of $\mathbf{r}$, say the m_1th element, instead of the whole vector. With this substitution, the *reconstructed* vector can be expressed as

$$\begin{aligned} \mathbf{r}_1 &= \boldsymbol{\mathcal{E}}_{m_1}\mathbf{r}_0 + \boldsymbol{\mathcal{I}}_{m_1}\widetilde{\mathbf{r}}_1 \\ &= \mathbf{r}_{noiseless} + \boldsymbol{\mathcal{I}}_{m_1}\widetilde{\mathbf{e}}_1 + \boldsymbol{\mathcal{E}}_{m_1}\mathbf{n} \end{aligned} \tag{4.7}$$

where we define $\mathbf{r}_0 \triangleq \mathbf{r}$, so that indices for subsequent iterations can be generalized easily. We defer how to maximize the benefit of this trade-off by choosing m_1 appropriately to Section 4.3.3.

For the first iteration ($i = 1$), we use the filter $\mathbf{G}$ to estimate $\mathbf{x}$ by using $\mathbf{r}_1$:

$$\begin{aligned} \widetilde{\mathbf{x}}_1 &= \mathbf{W}^{-1}\boldsymbol{\Gamma}^{-1}\mathbf{r}_1 \\ &= \mathbf{x} + \mathbf{W}^{-1}\boldsymbol{\Gamma}^{-1}\boldsymbol{\mathcal{I}}_{m_1}\widetilde{\mathbf{e}}_1 + \mathbf{W}^{-1}\boldsymbol{\Gamma}^{-1}\boldsymbol{\mathcal{E}}_{m_1}\mathbf{n} \\ &= \mathbf{x} + \mathbf{W}^{-1}\boldsymbol{\mathcal{I}}_{m_1}\mathbf{W}\mathbf{e}_0 + \mathbf{W}^{-1}\boldsymbol{\Gamma}^{-1}\boldsymbol{\mathcal{E}}_{m_1}\mathbf{n} \end{aligned} \tag{4.8}$$

since $\boldsymbol{\Gamma}^{-1}\boldsymbol{\mathcal{I}}_{m_1} = \boldsymbol{\mathcal{I}}_{m_1}\boldsymbol{\Gamma}^{-1}$ as both are diagonal matrices. This filtering is similar to the initial iteration in (4.4) except that $\mathbf{r}$, instead of $\mathbf{r}_1$, was used. The jth element of $\widetilde{\mathbf{x}}_1$ can be written as

$$\widetilde{x}_{1,j} = x_j + v_{1,j} + \sum_{m=1, m\neq m_1}^{M} \frac{\psi_{jm} n_m}{h_m}, \quad j = 1, 2, \cdots, M, \tag{4.9}$$

where $v_{1,j}$ is the jth element of $\mathbf{v}_1 \triangleq \mathbf{W}^{-1}\boldsymbol{\mathcal{I}}_{m_1}\mathbf{W}\mathbf{e}_0$. A decision function would yield the hard decided output as

$$\hat{\mathbf{x}}_1 = \mathrm{dec}(\widetilde{\mathbf{x}}_1), \tag{4.10}$$

hence completing the first iteration. For the second iteration and onwards, similar procedures are used as for the first iteration.

We summarize the ith iteration, $i = 1, 2, \cdots$, as

Algorithm 4.1 Iterative Subcarrier Reconstruction

- reconstruction: $\mathbf{r}_i = \boldsymbol{\mathcal{E}}_{m_i}\mathbf{r}_{i-1} + \boldsymbol{\mathcal{I}}_{m_i}\boldsymbol{\Gamma}\mathbf{W}\hat{\mathbf{x}}_{i-1}$;
- filtering: $\widetilde{\mathbf{x}}_i = \mathbf{G}\mathbf{r}_i$, where $\mathbf{G} = \mathbf{W}^{-1}\boldsymbol{\Gamma}^{-1}$;
- detection: $\hat{\mathbf{x}}_i = \mathrm{dec}(\widetilde{\mathbf{x}}_i)$.

We define a set containing all the indices that have been used for reconstruction as $\mathcal{M}_i = \{m_1, m_2, \cdots, m_i\}$. The complementary set of indices not yet used for reconstruction is denoted as $\mathcal{M}_i^C = \{m_{i+1}, m_2, \cdots, m_M\}$. In order to facilitate subsequent

[2]When multiplied with a vector, the matrix $\boldsymbol{\mathcal{E}}_m$ only *excludes* the mth element of the vector, while the matrix $\boldsymbol{\mathcal{I}}_m$ only *includes* the mth element.

derivations, in iteration i, we generalize the jth element of $\widetilde{\mathbf{x}}_i$ as

$$\widetilde{x}_{i,j} = x_j + v_{i,j} + \sum_{m \in \mathcal{M}_i^C} \frac{\psi_{jm} n_m}{h_m}, \quad j = 1, 2, \cdots, M. \tag{4.11}$$

4.3.3 Transform Design and Reconstruction Criteria

At this point, we have left the questions of which transform to use and the choice of $\mathcal{M}_i$ unanswered. To start, we assume that the elements of $\mathcal{M}_i$ are distinct elements. This means that for each reconstruction, one subcarrier not chosen previously is reconstructed. To select appropriate transform coefficients and $\mathcal{M}_i$ jointly, we require a sensible optimization criterion which optimizes the performance (in some sense) and admits a closed-form solution. We attempt to provide some possible answers in this section.

We make two simplifications to achieve our goal. We make the *error-free assumption* (EFA) where we assume that in iteration i, the decisions made in previous iterations are correct, i.e. $\mathbf{e}_{m-1} = 0$, $\forall m = 1, \cdots, i-1$. We also restrict the class of transforms to unitary transforms, which is appealing from the implementation and analysis point of view. Thus, we have

Rule 1. $\quad \boldsymbol{\Psi} = \mathbf{W}^H \Rightarrow \mathbf{W}^H \mathbf{W} = \mathbf{I}$.

This definition also ensures that the modulation symbol power σ_s^2 is normalized to the information symbol power σ_x^2, i.e.,

$$E[\mathbf{s}\mathbf{s}^H] = E[\mathbf{W}\mathbf{x}\mathbf{x}^H\mathbf{W}^H] = \sigma_x^2 \mathbf{I},$$

assuming that $E[\mathbf{x}\mathbf{x}^H] = \sigma_x^2 \mathbf{I}$. Note that $v_{i,j} = 0$ for $i = 0$, as there is no error propagation for the initial initialization. The EFA implies that $v_{i,j} = 0$ for any iteration $i \geq 1$. From (4.11), this allows us to obtain the SNR as

$$\gamma_{ij} = \frac{\bar{\gamma}}{\sum_{m \in \mathcal{M}_i^C} \frac{|w_{mj}|^2}{|h_m|^2}}. \tag{4.12}$$

where we have applied Rule 1. We denote $\bar{\gamma} = \sigma_x^2 / \sigma_n^2$ to be the average SNR.

The bit error rate (BER) would be limited by the worst performing symbol, especially at high SNR. Thus, we design the transform to maximize the minimum SNR γ_{ij} for $j = 1, 2, \cdots, M$. This is subject to the condition that the actual values of a given set of channel amplitudes $\{|h_m|^2\}$ are fixed but not known at the transmitter. Thus, the solution is robust in the sense that it does not depend on the channel statistics. Besides, since earlier iterations would lead to error propagation, the optimization should be carried out first for $i = 0$, then sequentially for increasing i. Mathematically, the transform, $\mathbf{W}$, and reconstruction indices, $\mathcal{M}_i$ are selected as

$$\arg \max_{\mathbf{W}, \mathcal{M}_i} \min_j \gamma_{ij} = \arg \min_{\mathbf{W}, \mathcal{M}_i} \max_j \sum_{m \in \mathcal{M}_i} \frac{|w_{mj}|^2}{|h_m|^2} \tag{4.13}$$

subject to Rule 1 (which constrains the transmission power) for $i = 0, 1, 2, \cdots$ in a sequential manner.

Consider the case when $i = 0$. In this case, the optimization is carried out for $\mathbf{W}$ only since γ_{0j} is not a function of $\mathcal{M}_i$. Since the channel amplitudes $\{|h_m|^2\}$ are not known at the transmitter, to satisfy (4.13), the maximin solution is given by $|w_{mj}|^2 = k^2$, where k^2 is a positive constant for $j, m = 1, 2, \cdots, M$. Otherwise, if $|w_{mj}|^2$ is not weighted uniformly, simply changing the indexing of the channel could lead to a smaller minimum SNR. Intuitively, having uniform $|w_{mj}|^2$ is robust against different channel conditions since all data symbols are transformed uniformly across the subcarriers. Thus, we may re-state this solution as

Rule 2. $\quad w_{nm} = \frac{1}{\sqrt{M}} e^{j\theta_{nm}}, \;\; n, m = 1, 2, \cdots, M.$

The set of arbitrary angles $\{\theta_{nm}\}$ is constrained because of Rule 1[3].

In [34], Rule 2 is shown to maximize the minimum BER at high SNR when the LS detection is used. However, our result applies for any channel, while [34] considers only a Rayleigh fading channel. Note that the design significantly differs when a different detection method is employed, such as the ML detector [35]. In general, our transform may not lead to optimum performance when a different detector is used, and similarly a transform designed for ML detection may not be optimum for our purpose here.

With Rule 2, the SNR becomes uniform across subcarrier $j = 1, \cdots, M$ for the initial iteration $i = 0$. In fact, this property clearly holds also for iteration $i = 1, 2, \cdots$. That is, the SNR (as optimized for $i = 0$) can be written (independently of j) as

$$\gamma_{ij} = \frac{M\bar{\gamma}}{\sum_{m \in \mathcal{M}_i^C} |h_m|^{-2}}. \tag{4.14}$$

For iteration $i \geq 1$, we determine the reconstruction indices m_i using the same optimization criterion (4.13) substituted with (4.14); thus, we are required simply to maximize γ_{ij}. We define $g_m \in \{h_l, l = 1, \cdots, M\}, m = 1, \cdots, M$ such that each g_m corresponds uniquely to one h_m and $|g_1|^2 \leq |g_2|^2 \leq \cdots \leq |g_M|^2$. Therefore, $\{g_m\}$ is simply $\{h_m\}$ but re-ordered such that the first index corresponds to the channel that gives the smallest amplitude, the second index gives the second smallest amplitude, and so on. It is easy to see that the SNR in (4.14) is largest when $\mathcal{M}_i$ is chosen such that the smallest $|h_{m_i}|^2, m_i \in \mathcal{M}_i$, is selected. Therefore,

$$\max \gamma_{ij} = \frac{M\bar{\gamma}}{\sum_{m=i+1}^{M} |g_m|^{-2}} \triangleq \gamma_i, \;\; i = 0, 1, 2, \cdots, \tag{4.15}$$

which is still independent of j. In order to maximize the SNR after the initial iteration, we conclude that for iteration i, we should always reconstruct the weakest subcarrier that has not been previously chosen.

[3]This rule is violated by the OFDM system where the PT is equal to the identity matrix.

4.3.4 Flexibility in Transform Design

The given transform design has some flexibility. A total of M^2 angles $\{\theta_{nm}\}$ can be chosen to satisfy Rule 2, subject to the constraints of Rule 1. The choice can be made based on considerations such as to minimize implementation complexity, reduce peak-to-average power ratio (PAPR), or simply to harmonize with current standards. We describe two well known transforms here and some of their advantages.

4.3.4.1 Walsh Hadamard Transform (WHT)

If the PT matrix equals the WHT matrix (known as WHT-OFDM), the columns are orthogonal Walsh Hadamard spreading codes. It is well known that the WHT matrix of order M can be constructed by an iterative procedure:

$$\mathbf{W}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}; \mathbf{W}_{2^k} = \mathbf{W}_{2^{k-1}} \otimes \mathbf{W}_2 \tag{4.16}$$

where $\otimes$ denotes the Kronecker product. The WHT-OFDM system becomes the conventional MC-CDMA system when the spreading code size is the same as the number of subcarriers M [73,89]. The WHT-OFDM system also offers a lower PAPR as compared to an OFDM system [88].

4.3.4.2 FFT

If we set $\mathbf{W}$ to be the FFT matrix[4] instead, then we obtain the conventional SC-FDE system. This observation is made in [34] as well. The SC-FDE system can enjoy lower PAPR as compared to an OFDM system, depending on the PAPR of the original information symbols.

4.3.5 Algorithm Refinement

The algorithm is designed using a linear LS filter, followed by a hard decision function and one reconstruction for each iteration. The performance can be improved by using the following refinements in the actual implementation, as verified in the simulations conducted in Section 4.6.

4.3.5.1 Use of linear MMSE filter

Instead of the LS filter, the MMSE filter can be used, which is designed to minimize the MSE of the error vector after filtering. Under the EFA, the solution for the filter is given as:

$$\mathbf{G} = \mathbf{W}^{-1}\mathbf{B}_i \tag{4.17}$$

[4]We assume that the FFT is normalized to fulfil Rule 1.

where $\mathbf{B}_i$ is a diagonal matrix defined in Appendix 4.A. Appendix 4.A shows that the transform design and reconstruction criteria derived for the LS filter are still valid when the MMSE filter is used instead. It is to be noted that for weak subcarriers with small $|h_m|$, $|b_m|$ is small to avoid the excessive noise amplification. However, for strong subcarriers with large $|h_m|$, b_m becomes closer to $1/h_m$ which reduces interference due to loss of orthogonality. Therefore the MMSE detector compromises between effects of excessive noise amplification and interference due to loss of orthogonality, and thus outperforms the LS detector in most communication systems.

4.3.5.2 Use of soft decision function

The decision function at each iteration can be altered to improve the performance. For example, the clipping function for BPSK signals may be given as

$$\widehat{x} = \begin{cases} -x_R, & \widetilde{x} \leq -x_R, \\ \widetilde{x}, & |\widetilde{x}| < x_R, \\ +x_R, & \widetilde{x} \geq x_R. \end{cases} \tag{4.18}$$

Here, x_R can be set for instance as $x_R = 1$ if the BPSK signals are normalized to unit power. For QPSK signals, the clipping function in (4.18) may be applied to the real and imaginary parts separately. By using the clipping function, a hard decision is made only for those bits with more energy which are relatively robust to noise. For bits with less energy and thus more susceptible to an erroneous decision, decisions are not made with the hope that further iterations would assist in improving decisions made in the last iteration.

4.3.5.3 Iterate using the same reconstruction subcarrier

The iterative process can also be applied to one particular subcarrier a few times for the effects of the reconstruction to be fully realized before the next subcarrier is reconstructed. Based on simulation results, some improvement is obtained for the first three to four repetitions, after which the amount of subsequent improvement becomes negligible.

4.4 Performance Analysis under EFA

In this section, we consider the derivation of the upper and lower BER bounds under EFA. The lower BER bound under the EFA is the true lower bound of the actual BER. The derivation of the upper bound under EFA is a by-product of the analysis. However, with knowledge of the upper bound, we can determine the diversity of the iterative detector under EFA. Practically, the EFA is never valid but it allows us to understand the behavior of the iterative detector under optimistic conditions. Also, it

is expected that the EFA can be well approximated when channel codes are introduced and the output of the channel decoder is fed back to the detector.

We obtain the BER bounds by first bounding the SNR appropriately using an LS filter. The objective is to obtain a closed-form expression which produces insights on the performance of the system. Similar derivations for the BER can be applied using an MMSE filter; however, since only the SINR can be calculated directly, where the interference is non-Gaussian, the BER obtained via the SINR serves only as an approximation.

We can bound the SNR γ_i (4.15) for the ith iteration as

$$\gamma_{\text{low},i} \leq \gamma_i \leq \gamma_{\text{up},i}, \tag{4.19}$$

where the upper bound is given by

$$\gamma_{\text{up},i} \triangleq M\bar{\gamma}|g_{i+1}|^2, \tag{4.20}$$

and the lower bound is given by

$$\begin{aligned} \gamma_{\text{low},i} &\triangleq M\bar{\gamma}|g_{i+1}|^2/(M-i) && (4.21)\\ &= \gamma_{\text{up},i}/(M-i). && (4.22) \end{aligned}$$

The bounds in (4.19) follow because we have chosen the subcarriers for reconstruction such that $|g_{i+1}| \leq |g_{i+2}| \leq \cdots \leq |g_M|$. Since the lower bound $\gamma_{\text{low},i}$ is simply the upper bound $\gamma_{\text{up},i}$ scaled by $(M-i)$, the BER based on lower bound $\gamma_{\text{low},i}$ can be obtained easily once the BER based on upper bounds $\gamma_{\text{up},i}$ is known. In addition, the SNR difference between both bounds is $10\log 10(M-i)$, which improves as M gets smaller or as i gets larger.

For a single-antenna system where no receive diversity can be obtained, we assume that $h_1, h_2, \ldots, h_M$ are i.i.d. Gaussian distributed. The independence assumption is valid if we select adjacent subcarriers that are spaced more than the coherence bandwidth in frequency to form a group, and in this case the model presented in (4.1), (5.1) is used for the individual groups. For an $L \geq 2$ receive antenna system, we treat $\mathbf{\Gamma}$ as the equivalent channel after maximal ratio combining (MRC) is applied. Thus, in general $|h_1|^2, |h_2|^2, \ldots, |h_M|^2$ are modelled as i.i.d. central chi-square distributed random variables with $2L$ degrees of freedom where $E[|h_m|^2] = L$ for all m. For the case when channel diversity is provided by the multiple transmit antennas through orthogonal space time block codes [90], the analysis below can be applied similarly since the channel statistics are the same except for a simple variable change to account for differences in the mean of the SNR.

4.4.1 BER Bounds under EFA

In order to bound the BER, we need to obtain the probability density function (PDF) of $\gamma_{\text{low},i}$ and $\gamma_{\text{up},i}$. This is done by first considering, in general, the distribution of

the ordered SNR $\gamma_i = \alpha|g_i|^2$, denoted as $f_{\gamma_i}(\gamma)$. The results are given as (4.38) and (4.39) for the case of $L = 1, 2$, respectively, in Appendix 4.B. The extensions of the results for $L \geq 3$ can be carried out similarly. The PDF of the lower-bound SNR $\gamma_{\text{low},i} = \frac{M\bar{\gamma}}{M-i}|g_i|^2$, can then be obtained easily from that of $f_{\gamma_i}(\gamma)$ by substituting α with $\bar{\gamma}_{\text{low},i} \triangleq \frac{M\bar{\gamma}}{M-i}$. Similarly, the PDF of $\gamma_{\text{up},i} = M\bar{\gamma}|g_i|^2$, can be obtained by substituting α with $\bar{\gamma}_{\text{up},i} \triangleq M\bar{\gamma}$.

4.4.1.1 Conventional OFDM System

We first consider the BER of an OFDM system with an Lth order diversity channel as a benchmark for comparison with a PT-OFDM system. Let the BER of a modulated system in an AWGN channel be $P_{\text{awgn}}(\gamma)$. With the PDF of the SNR given in (4.35), the BER of an Lth order diversity channel is

$$P_L^{\text{fading}}(\bar{\gamma}) = \int_0^\infty P_{\text{awgn}}(\gamma) f_\gamma(\gamma)\, d\gamma. \tag{4.23}$$

For common modulation schemes such as MQAM and MPSK schemes, closed-form expressions or expressions amenable to numerical calculation are available [75, 91]. We consider the specific case of a QPSK modulated system, where the closed-form expression for the BER is

$$P_L^{\text{fading}}(\bar{\gamma}) = \left(\frac{1-\mu}{2}\right)^L \sum_{l=0}^{L-1} \binom{L-1+l}{l} \left(\frac{1+\mu}{2}\right)^l \tag{4.24}$$

where $\mu = \sqrt{\frac{\bar{\gamma}}{1+\bar{\gamma}}}$.

4.4.1.2 PT-OFDM System

Now, we are ready to consider the BER for the iterative PT-OFDM system. Using the SNR γ_i defined in (4.15), the exact BER for the ith iteration is given by a multi-fold integration:

$$P_{\text{e}}(i, \bar{\gamma}) = \int_{\mathbf{g}_i} P_{\text{awgn}}(\gamma_i|\mathbf{g}_i) f_{\mathbf{g}_i}(\mathbf{g}_i)\, d\mathbf{g}_i,\ i = 0, 1, \cdots, M-1, \tag{4.25}$$

where we define $\mathbf{g}_i = \left[|g_{i+1}|^2, |g_{i+2}|^2, \cdots, |g_M|^2\right]^T$, and $f_{\mathbf{g}_i}$ is its multivariate PDF. A closed-form expression cannot be obtained easily for the general case, not even for the initial iteration where $i = 0$. However, since the instantaneous SNR is lower bounded by $\gamma_{\text{low},i}$ in (4.19), the instantaneous BER can be upper bounded. Similarly, since the instantaneous SNR is upper bounded by $\gamma_{\text{up},i}$, the instantaneous BER can be lower bounded. Taking the expectation of the instantaneous BER, we are then

able to bound the average BER in an Lth order diversity channel. The BER is upper bounded as follows:

$$P_{\mathrm{e}}(i,\bar{\gamma}) \leq P_{\mathrm{up}}(i,\bar{\gamma}) \triangleq \int_0^{\infty} P_{\mathrm{awgn}}(\gamma) f_{\gamma_{\mathrm{low},i}}(\gamma)\, d\gamma. \tag{4.26}$$

Here, only a single-fold integration needs to be carried out. Moreover, a closed-form expression can be found for $L = 1$ and 2 for QPSK modulated symbols, as follows.

Consider $L = 1$. We substitute (4.38) into (4.26) and let $\alpha = \bar{\gamma}_{\mathrm{low},i}$. After some manipulations, we obtain the upper-bound BER as

$$P_{\mathrm{up}}(i,\bar{\gamma}) = \frac{M!}{i!(M-i-1)!} \sum_{k=0}^{i} \binom{i}{k} \frac{(-1)^k}{\beta} P_1^{\mathrm{fading}}\left(\frac{\bar{\gamma}_{\mathrm{low},i}}{\beta}\right) \tag{4.27}$$

where $\beta = M - i + k$. This explicit formulation of the BER in terms of (4.23) is advantageous since we can make use of the rich literature that investigates (4.23) for different scenarios. In this case, closed-form expressions exist for many common modulation schemes, such as for QPSK as given in (4.24), and thus the upper-bound BER has a closed-form expression for such cases as well. The lower bound BER, denoted as $P_{\mathrm{low}}(i,\bar{\gamma})$, can be derived similarly by using $\gamma_{\mathrm{up},i}$. Alternatively, note that since the upper-bound and lower-bound SNR differ by a factor of $M - i$, we can immediately derive it as

$$P_{\mathrm{low}}(i,\bar{\gamma}) = P_{\mathrm{up}}\left(i,(M-i)\bar{\gamma}\right). \tag{4.28}$$

Next, consider $L = 2$. Substituting (4.39) into (4.26) and letting $\alpha = \bar{\gamma}_{\mathrm{up},i}$, and after similar manipulations, we can also obtain the upper-bound BER for iteration i in terms of (4.23):

$$P_{\mathrm{up}}(i,\bar{\gamma}) = \frac{M!}{i!(M-i-1)!} \sum_{k=0}^{i} \binom{i}{k} \sum_{n=0}^{\beta-1} \binom{\beta-1}{n} \frac{(-1)^k (n+1)!}{\beta^{n+2}} P_{n+2}^{\mathrm{fading}}\left(\frac{\bar{\gamma}_{\mathrm{low},i}}{\beta}\right) \tag{4.29}$$

and the lower-bound BER is

$$P_{\mathrm{low}}(i,\bar{\gamma}) = P_{\mathrm{up}}\left(i,(M-i)\bar{\gamma}\right). \tag{4.30}$$

For $L > 2$, similar derivations can be carried out with an increase in the number of summation operators.

We formally summarize the main results in the following theorem:

Theorem 4.1. *Consider the PT-OFDM system that employs iterative detection with SNR in iteration i given by γ_i in (4.15). We make the error-free assumption (EFA) for iteration $i \geq 1$. (This assumption is not necessary for $i = 0$.) Then, the BER for the ith iteration is bounded as*

$$P_{low}(i,\gamma) \leq P_e(i,\gamma) \leq P_{up}(i,\gamma)$$

where $P_{up}(i,\gamma)$ and $P_{low}(i,\gamma)$ are given by (4.27) and (4.28), respectively, for $L = 1$; and given by (4.29) and (4.30), respectively, for $L = 2$.

4.4.2 Performance Comparison with Conventional Scheme

In this section, we make an analytical comparison with the conventional scheme. In some cases, namely when a simple comparison is not possible, asymptotic results are presented.

In order for us to derive some useful results, we assume that the BER of Lth order diversity channel given by (4.23) can be expressed as

$$P_L^{\text{fading}}(\bar{\gamma}) = \sum_{j=L}^{\infty} \frac{a_j}{\bar{\gamma}^j} \tag{4.31}$$

for some constants $a_j, j = L, L+1, \cdots$. This is shown to be possible for a QPSK modulated system in Appendix 4.C for $L = 1, 2$. In general, it is also easy to show that (4.31) is valid for MQAM and QPSK modulated systems for any integer L.

4.4.2.1 Initial Iteration, $i = 0$

For the initial iteration $i = 0$, Theorem 4.1 does not rely on the EFA. Thus, Theorem 4.1 holds true for a variety of systems such as MC-CDMA or SC-FDE system where no iteration is carried out, and is thus useful in its own right. If no iteration is performed and LS detection is employed, then we should always use a conventional OFDM system instead of a PT-OFDM system. Corollary 4.1 explains why this is so. All proofs for the corollaries are given in Appendix 4.D.

Corollary 4.1. *We make the same assumptions as in Theorem 4.1. Let $P_e(0, \bar{\gamma})$ be the exact BER of the PT-OFDM system when no iteration is performed, and let $P_1^{fading}(\bar{\gamma})$ be the BER of the conventional OFDM system in a single-antenna system. Then $P_e(0, \bar{\gamma}) \geq P_1^{fading}(\bar{\gamma})$.*

In words, if iteration is not performed (with $i = 0$) and the channel diversity order is limited (with $L = 1$), a conventional OFDM system performs at least as well as the PT-OFDM system in terms of BER.

For a system with Lth order channel diversity where $L > 1$, Corollary 4.1 does not hold in general. Instead, motivated by the simulation results that the exact BER approaches the lower-bound BER at high SNR for $L = 2$, we present the following corollary to explore the asymptotic performance.

Corollary 4.2. *We make the same assumptions as in Theorem 4.1. Assume further that (4.31) is valid. Then, $P_{low}(0, \bar{\gamma}) \rightarrow \frac{1}{M} P_2^{fading}(\bar{\gamma})$ as SNR$\rightarrow \infty$.*

In words, if iteration is not performed (with $i = 0$) but there is a channel diversity order of two (with $L = 2$), the PT-OFDM system has a lower-bound BER that is M times smaller than that of a conventional OFDM system at high SNR. With Corollary 4.2 and backed by the simulation results shown in Section 4.6, we can conclude that the use of a pre-transform is able to exploit the channel diversity to achieve better performance when M is large and the SNR is sufficiently high.

4.4.2.2 Diversity Advantage in Iterative Detection of PT-OFDM

For $L = 1$, Corollary 4.1 shows that without further iterations the performance of the PT-OFDM system is limited. In this case, Corollary 4.3 provides the theoretical motivation for using the proposed iterative method, in terms of the diversity advantage. The diversity advantage is defined as

$$\lim_{\bar{\gamma}\to\infty} \frac{-\log(\mathrm{BER})}{\log(\bar{\gamma})}$$

which measures the asymptotical rate of decrease of the BER as the average SNR, $\bar{\gamma}$, increases.

Corollary 4.3. *We make the same assumptions as in Theorem 4.1. Then, for $L = 1$, the iterative detector provides a diversity advantage of at least $i + 1$ after completing iteration i.*

4.5 Other Issues

4.5.1 Complexity

We next consider the complexity of the algorithm, measured as the number of required complex multiplications and divisions. We neglect the complexity of the detection stage since it is relatively simple to implement. In addition, we do not make any assumption on the elements of $\mathbf{W}$ and $\hat{\mathbf{x}}_i$. Further complexity reduction can be made, for example, when the symbols are QPSK modulated or when a well-structured matrix such as the FFT matrix is used.

Consider $i = 0$. The reconstruction stage is not required. The computation of the filtering stage is $\widetilde{\mathbf{x}}_0 = \mathbf{G}\mathbf{r}$, whose complexity is reflected in Table 4.1.

The complexity of the iterative subcarrier reconstruction algorithm, as shown in Algorithm 4.1, can be further reduced. By using the fact that $\boldsymbol{\mathcal{E}}_m = \mathbf{I} - \boldsymbol{\mathcal{I}}_m$, it can be shown that we can alternatively implement the algorithm recursively for $i \geq 1$ as Algorithm 4.2:

Algorithm 4.2 Iterative Subcarrier Reconstruction – Alternative Version

- reconstruction: $\mathbf{r}_i = \mathbf{r}_{i-1} - \mathbf{c}_i$, where $\mathbf{c} = \boldsymbol{\mathcal{I}}_{m_i}\left(\mathbf{r}_{i-1} - \boldsymbol{\Gamma}\mathbf{W}\hat{\mathbf{x}}_{i-1}\right)$;
- filtering: $\widetilde{\mathbf{x}}_i = \widetilde{\mathbf{x}}_{i-1} - \mathbf{G}_i\mathbf{c}_i + \mathbf{d}_i$,
- detection: $\hat{\mathbf{x}}_i = \mathrm{dec}(\widetilde{\mathbf{x}}_i)$.

In the filtering step, if the LS filter is used we have

$$\begin{aligned} \mathbf{G}_i &= \mathbf{W}^H\boldsymbol{\Gamma}^{-1}, \\ \mathbf{d}_i &= \mathbf{0}, \end{aligned}$$

iteration i	Iterative LS detector	Iterative MMSE detector
0	$M^2 + M(\times), M(\div)$	$M^2 + 2M(\times), M(\div)$
≥ 1	$2M + 2(\times)$	$3M + 2(\times), 1(\div)$
Total for I iterations	$M^2 + M + I(2M + 2)(\times)$ $M(\div)$	$M^2 + 2M + I(3M + 2)(\times)$ $M + I(\div)$

Table 4.1: Complexity of the proposed iterative detector in terms of multiplications ($\times$) and divisions ($\div$).

while if the MMSE filter is used, we have instead

$$\begin{aligned} \mathbf{G}_i &= \mathbf{W}^H \mathbf{B}_i, \\ \mathbf{d}_i &= \mathbf{W}^H \left(\frac{1}{h_{m_i}} - \frac{h_{m_{i-1}}}{|h_{m_{i-1}}|^2 + \sigma_n^2} \right) \boldsymbol{\mathcal{I}}_{m_i} \mathbf{r}_{i-1}, \end{aligned}$$

where $\mathbf{B}_i$ is defined in Appendix 4.A. By reusing previous calculated results and performing only necessary operations (for instance when $\boldsymbol{\mathcal{I}}_m$ and diagonal matrices are involved), the complexity is obtained as shown in Table 4.1.We omit the lengthy but straightforward calculations to obtain the numbers. Some minor differences are expected for different implementation; however, the order of the complexity would not be affected.

Table 4.1 shows the complexity obtained for a total of I iterations, which include repetitions on the same subcarrier in the reconstruction step. We observe that for $i = 0$, the complexity is quadratic in M, while for $i \geq 1$, the complexity is linear in M. This means that much of the complexity comes from the initial iteration, which is required even for conventional linear detectors. Each iteration, on the other hand, adds only a marginal complexity.

4.5.2 Imperfect Knowledge of Channel

In practical implementations, the channel is not known exactly at the receiver but it needs to be estimated via a training sequence. The iterative detection schemes is not particularly sensitive to channel estimation errors. When an LS channel estimator is used for simulations, it shows the same amount of degradation as experienced by conventional OFDM and PT-OFDM systems. Thus, the same performance superiority is maintained over conventional systems.

4.5.3 Coded Performance

The iterative detector can be concatenated with a channel code to further improvement its performance. Depending on the codes used, it can lead to a varying degree of coding gain. The iterative detector can be further optimized with respect to the channel codes as well.

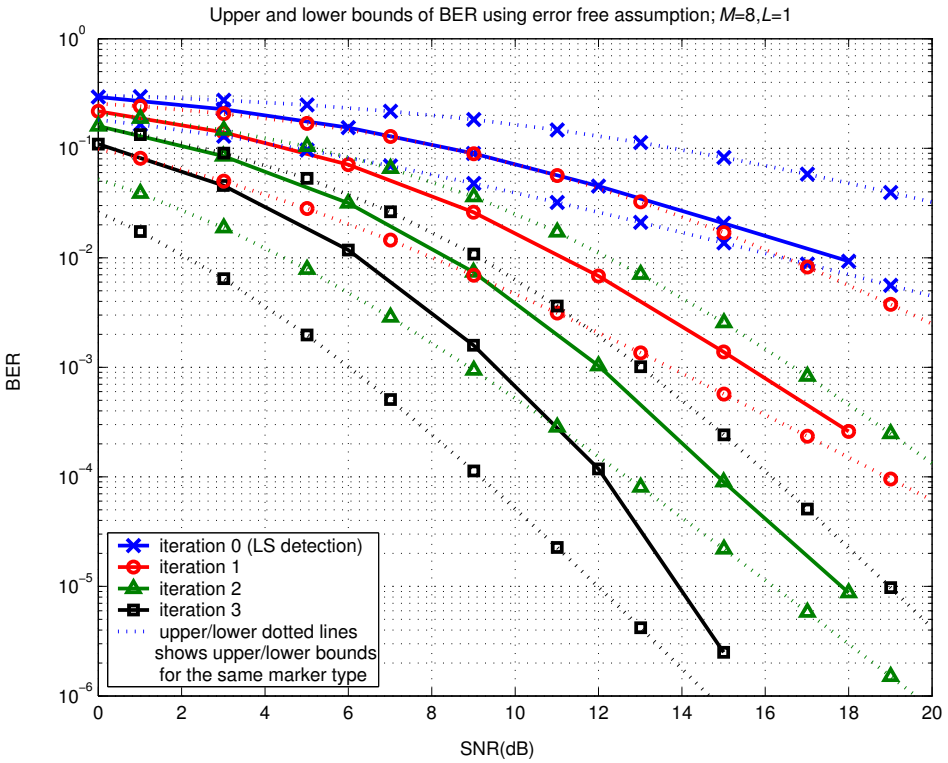


Fig. 4.2: BER (full lines) using iterative subcarrier reconstruction in a WHT-OFDM system with $M = 8, L = 1$. Analytical upper and lower BER bounds are indicated by upper and lower dotted lines, respectively.

4.6 Simulation Results

4.6.1 Performance under EFA and Independent Subcarriers

In order to verify and reinforce the conclusions made from the analysis in Section 4.4, we perform simulations for the PT-OFDM system under the EFA. We emphasize that the (exact) BER *with EFA* considered (and hence its lower bound) serves as a lower bound for the BER *without EFA*. However, the BER with EFA is useful to highlight the mechanism and potential benefit of the algorithm. In our simulations, the number of time domain channel taps is equal to the size of the matrix and we assume a uniform power delay profile (i.e. i.i.d. channels for the subcarriers). We use WHT for the transform.

For $L = 1$, as a test case, we set $M = 8$ and $i = 0, 1, 2, 3$. We simulate the performance of the QPSK modulated WHT-OFDM system where we made the EFA. The results are given in Fig. 4.2. It is seen that the simulated BERs fall between their respective

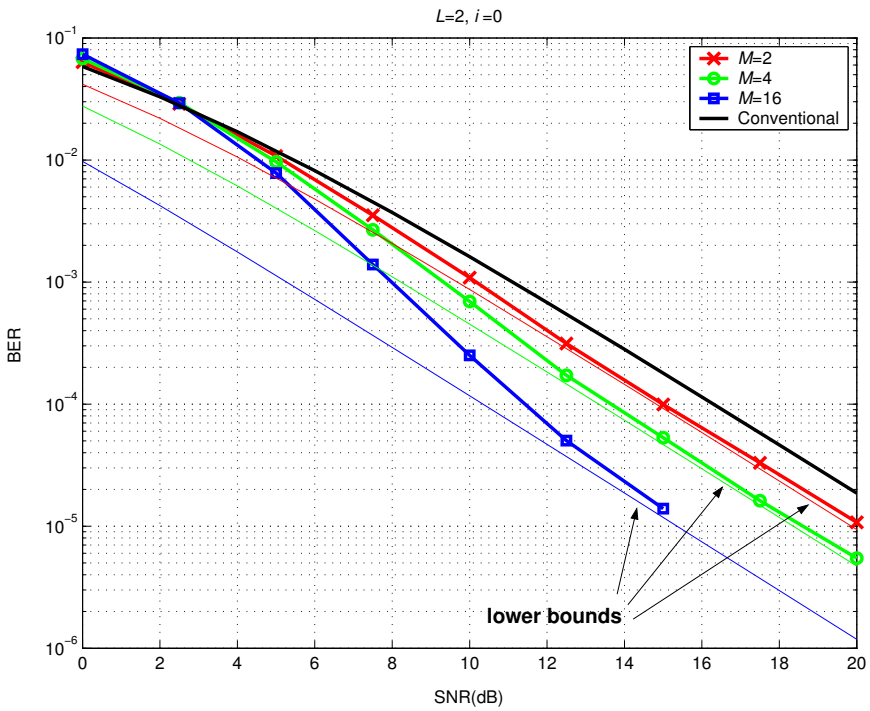


Fig. 4.3: BER using iterative subcarrier reconstruction in a WHT-OFDM system with $L = 2$ and $M = 2, 4, 16$. The lower bounds are plotted under the error-free assumption (EFA).

upper and lower bounds predicted in Theorem 4.1. It is also verified from simulations that the conventional OFDM system has the same BER as the lower bound of PT-OFDM for $i = 0$ (not shown in the figure for the sake of clarity), thus validating Corollary 4.1. Lastly, it appears that the BERs converge to the lower bound at high SNR, which has a diversity advantage of $i+1$ at iteration i. This agrees with Corollary 4.3. Hence, although the upper and lower bounds of the BER can vary by as much as 6 dB at high SNRs, both bounds (being the same) serve to confirm the diversity order of the BER.

For $L = 2$, we use $M = 2, 4, 16$ and $i = 0$. The simulation result is shown in Fig. 4.3. It is seen that increasing M would be sufficient to increase the performance of a PT-OFDM system when combined with diversity techniques. At high SNR (> 14dB), it is seen that the simulated BER starts to approach the analytical lower bound, as stated in Corollary 4.2.

4.6.2 Performance in Practical Scenarios

In this section, we present simulation results of the PT-OFDM system in a practical frequency-selective fading channel. We assume that the duration of cyclic prefix is the

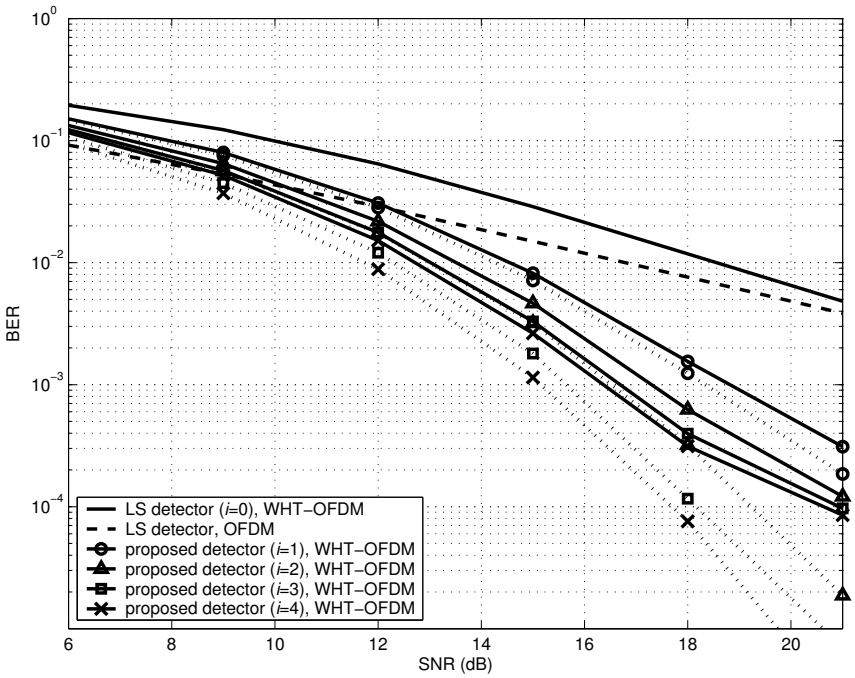


Fig. 4.4: BER using iterative subcarrier reconstruction (see proposed detector after i iterations) in a WHT-OFDM system with $M = 64, L = 1$. The dotted lines show the BER bounds under EFA. Here, ZF equalization with hard decision is employed in each detection step.

same as the maximum delay spread. The channel is modelled as an order $L_h = 16$ FIR filter with sampled, truncated exponential power delay profile and root mean squared delay spread $\tau_{rms} = 4$ (normalized to the sample interval). The number of subcarriers is set to $M = 64$. For each iteration, 4 reconstructions are repeated on the same subcarrier to ensure convergence.

Fig. 4.4 illustrates the BER performance of the iterative LS detectors of a PT-OFDM system with hard decision. For comparison, we also show the BER performance of a conventional LS detector, for both PT-OFDM and OFDM systems. We see that the iterative detector improves the BER performance significantly especially at high SNR, as opposed to using the LS detector in the PT-OFDM and OFDM systems. For BER at 10^{-2}, the iterative algorithm using the four worst subcarriers for reconstruction outperforms the PT-OFDM and OFDM systems by 6 dB and 4 dB, respectively. Even with only one replacement, it outperforms its counterparts by $4 - 4.5$ dB and $2 - 2.5$ dB respectively. As more subcarriers are replaced, the performance margin diminishes. Therefore replacing a few worst subcarriers, which most influence overall performance, is sufficient to obtain the desired performance improvement and this makes the iterative detector easy to implement.

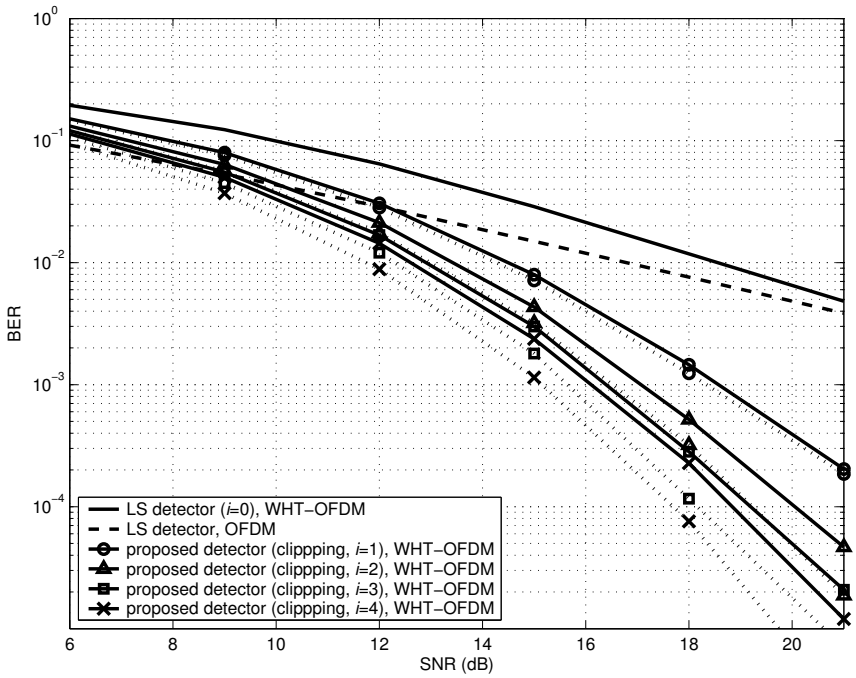


Fig. 4.5: BER using iterative subcarrier reconstruction (see proposed detector after i iterations) in a WHT-OFDM system with $M = 64, L = 1$. The dotted lines show the BER bounds under EFA. Here, ZF equalization with clipping function is employed in each detection step.

The dotted lines in Fig. 4.4 show the simulated performance curves under EFA, which act as lower bounds for the actual performance curves without EFA. We see that there is a gap between both curves. This gap can be reduced partly by using a clipping function instead of a hard decision in each iteration. This is shown in Fig. 4.5. The improvement for the BER at 10^{-3} is trivial compared with using hard decision. At higher SNR, the improvement is more significant. It can be seen that the performance of the proposed algorithm approximates the EFA bound well especially for the first a few iterations.

When the SNR is known at the receiver, significantly better performance can be obtained by using an MMSE filter. Fig. 4.6 illustrates the BER performance of the iterative MMSE filter based detector. When $i = 0$, the iterative MMSE detector is simply the classical MMSE filter. At BER of 10^{-4}, approximately 2.5 dB of performance gain is achieved using six iterations over the classical MMSE detector. The performance gain is even more significant when it is compared to the LS detector for an OFDM system.

Next, we compare the performance shown in the previous figures with the PIC detector which is commonly used for multi-user detection [37,38]. The initial stage of the PIC is the same as the initial iteration of the proposed iterative detector. The initialization

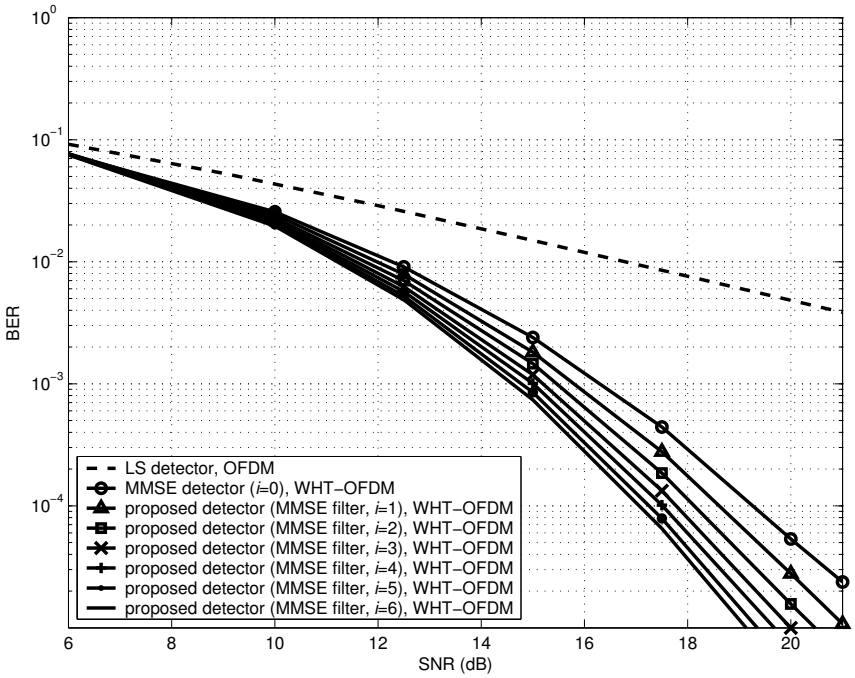


Fig. 4.6: BER using iterative subcarrier reconstruction (see proposed detector after i iterations) in a WHT-OFDM system with $M = 64, L = 1$. The dotted lines show the BER bounds under EFA. Here, MMSE equalization with hard decision is used in each detection step.

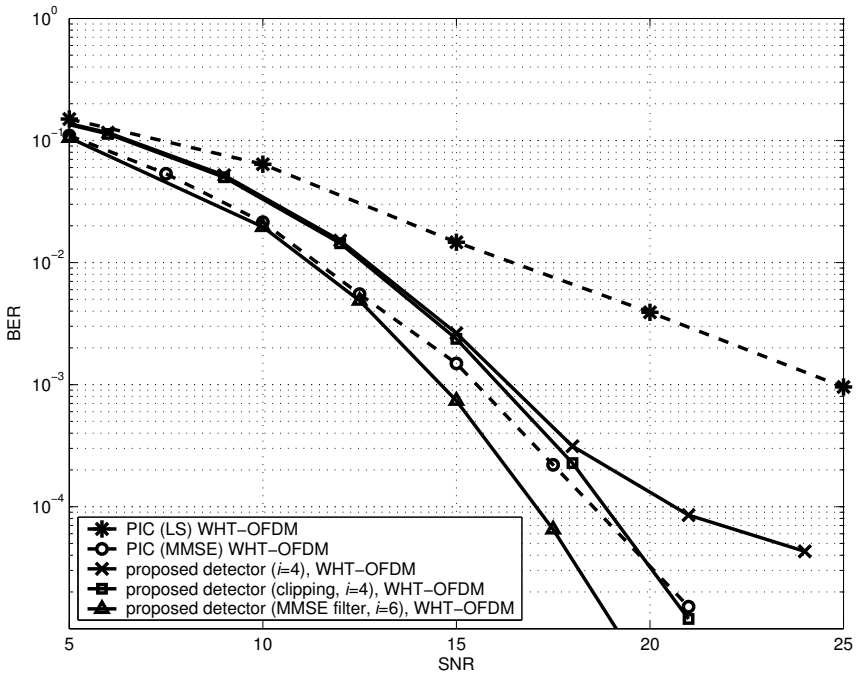


Fig. 4.7: BER using iterative subcarrier reconstruction compared to parallel interference cancelation schemes.

can be carried out either by an LS or MMSE detector. Subsequently, the ith stage of the PIC is given as

$$\widetilde{x}_{i,m} = \mathbf{h}_m^H \left(\mathbf{r} - \mathbf{\Gamma W} \boldsymbol{\mathcal{E}}_m \widehat{\mathbf{x}}_{i-1} \right), m = 1, 2, \cdots, M \tag{4.32}$$

where $\mathbf{h}_m$ is the mth column of the matrix $\mathbf{\Gamma W}$. The decision of the mth symbol at the ith stage is carried out as $\widehat{x}_{i,m} = \mathrm{dec}(\widetilde{x}_{i,m})$ where a hard decision is applied. We observed in simulations that the performance becomes worse after the first iteration, so only the performance at the first iteration is shown in Fig. 4.7. It is seen that the performance of the PIC is worse than the proposed iterative detector when both use the LS filter, and similarly when both use the MMSE filter.

We consider the complexity of the PIC detector next. Calculating $\mathbf{\Gamma W} \boldsymbol{\mathcal{E}}_m \widehat{\mathbf{x}}_{i-1}$ requires M^2 complex multiplications and multiplying with $\mathbf{h}_m^H$ requires another M multiplications. Thus, calculating $\widetilde{x}_{i,m}$ requires $M^2 + M$ multiplications for symbol m. To detect all symbols, one stage of the PIC would then require $M(M^2 + M)$ multiplications, which is of cubic order of complexity. This is substantially more complex than the proposed iterative detector, since even when 6 iterations are carried out, the complexity is still very low as demonstrated in Section 4.5.

Fig. 4.8 makes the comparison of the proposed iterative detectors with an ML detector designed in [35]. For ML detection, unitary square transforms of size two, four and

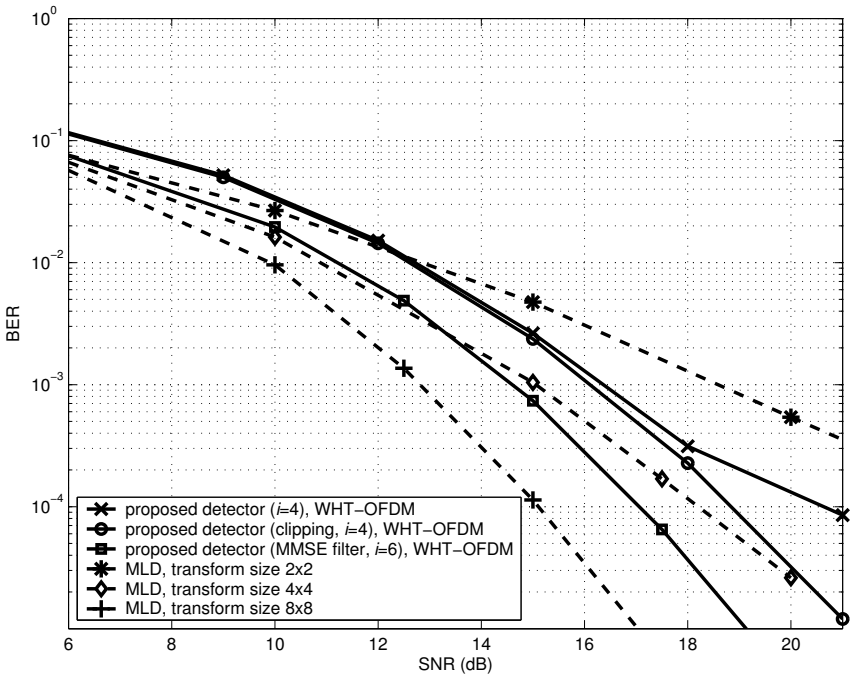


Fig. 4.8: BER using iterative subcarrier reconstruction compared to MLD detection.

eight are used. The transformation is carried out over groups of subcarriers, whereby the subcarriers are selected such that the subcarrier indices are spaced as far apart as possible, as described in [35]. It is seen in the simulations that the performance for the ML detector depends on the size of the transform and could be better than using our proposed detector. However, the complexity of implementation is very high, namely in the order of $O(4^M)$ when QPSK signals are used, and may be unacceptable. The order of complexity of the iterative detector, on the other hand, is quadratic as shown in Section 4.5.1.

4.7 Conclusion

In this chapter, a class of low-complexity iterative detectors is proposed for a PT-OFDM system. Such a detector reduces the overall system noise by iteratively trading signals at weak subcarriers with estimated ones. We obtain the design rules for such a system based on the objective of maximizing the minimum SNR across subcarriers. These rules give rise to a family of transforms. The rules also dictate that the signal at the weakest subcarrier should be traded first, followed by the second weakest, and so on. Analytically, we show that the proposed iterative detector leads to an increased diversity advantage at high SNR if we may assume that the previous detections are

correct. Finally, simulations in practical wireless channels demonstrate the superior performance of the iterative detector as compared to conventional linear detectors. For simplicity, we have used the LS detector and a hard decision device to obtain our design rules and perform the analysis. Simulations show that more advanced techniques, such as based on the MMSE detector and a soft-decision device, can further improve the performance.

Appendix 4.A Considerations for the MMSE filter

In this appendix, we consider the derivation of the MMSE filter and the signal-to-(interference plus noise) ratio (SINR) under the EFA. We then derive the transform design and reconstruction criteria by maximizing the minimum SINR. It is shown that by making the same simplifications as for the LS filter (i.e. Rule 1 and the EFA), we reach the same solution.

With no error propagation, we have $\hat{\mathbf{x}}_i = \mathbf{x}$ and from Section 4.3.2, we can simplify the received vector at iteration i as

$$\mathbf{r}_i = \mathbf{\Gamma W x} + \widetilde{\mathbf{n}} \tag{4.33}$$

where

$$\widetilde{n}_m = \begin{cases} 0, & m \in \mathcal{M} \\ n_m, & m \in \mathcal{M}^C. \end{cases}$$

Without loss of generality, we assume that $\sigma_x^2 = 1$. Defining $\mathbf{H} = \mathbf{\Gamma W}$ and $\mathbf{R}_{\widetilde{\mathbf{n}}} = E[\widetilde{\mathbf{n}}\widetilde{\mathbf{n}}^H]$, by using standard derivations we obtain the MMSE filter as

$$\mathbf{G} = \left[\mathbf{H}^H\mathbf{H} + \mathbf{R}_{\widetilde{\mathbf{n}}}\right]^{-1}\mathbf{H}^H.$$

Upon simplification, we obtain

$$\mathbf{G} = \mathbf{W}^H\mathbf{B}_i \tag{4.34}$$

where $\mathbf{B}_i = \mathrm{diag}(b_1, b_2, \cdots, b_M)$,

$$b_m = \begin{cases} h_m^{-1}, & m \in \mathcal{M}_i \\ h_m^*/(|h_m|^2 + \sigma_n^2), & m \in \mathcal{M}_i^C \end{cases}$$

and σ_n^2 is the noise power.

Next, we consider the SINR of the signal after MMSE filtering, $\widetilde{\mathbf{x}}_i = \mathbf{G}\mathbf{r}_i$. After some algebra, we obtain

$$E[\widetilde{\mathbf{x}}_i\widetilde{\mathbf{x}}_i^H] = \mathbf{W}^H\mathbf{B}_i\mathbf{\Gamma W}.$$

Thus, the sum of the signal, interference and noise powers of the jth symbol is given by the diagonal element of the matrix as $E[|\widetilde{x}_{i,j}|^2] = \sum_{m=1}^{M}|w_{jm}|^2 h_m b_m \triangleq \delta_j > 0$. Also, after filtering, the signal component of the jth symbol can be extracted from $\widetilde{\mathbf{x}}_i = \mathbf{W}^H\mathbf{B}_i(\mathbf{\Gamma W x} + \widetilde{\mathbf{n}})$ by observation as $\mathbf{w}_j^H\mathbf{B}_i\mathbf{\Gamma}\mathbf{w}_j x_j = \sum_{m=1}^{M}|w_{jm}|^2 h_m b_m x_j$. The signal power is therefore δ_j^2 and the SINR is given as

$$\mathrm{SINR} = \frac{\delta_j^2}{\delta_j - \delta_j^2} = \frac{1}{\delta_j^{-1} - 1}.$$

Following the same procedures used for LS filter, we seek the solution for $\mathbf{W}$ and $\mathcal{M}_i$ by maximizing the minimum SINR over different j. By the same argument, we

deduce that Rule 2 is required for $i = 0$. This resulted in a uniform SINR across all j, regardless of the choice of $\mathcal{M}_i$. Then, we can maximize the SINR according to

$$\begin{aligned} \max \text{SINR} &= \max \delta_j \\ &= \max \sum_{m \in \mathcal{M}_i^C} \frac{|h_m|^2}{|h_m|^2 + \sigma_n^2} + |\mathcal{M}_i| \\ &= \max \sum_{m \in \mathcal{M}_i^C} \frac{1}{1 + \sigma_n^2/|h_m|^2}, \end{aligned}$$

where $|\mathcal{M}_i|$ is the cardinality of $\mathcal{M}_i$. Thus, for the SINR to be maximized, we need to choose $\mathcal{M}_i$ such that it corresponds to the set of indices corresponding to i smallest channel amplitudes. This resulted in the same solution as for the case when the LS filter is used.

Appendix 4.B Derivation of the PDF of $\gamma_i = \alpha|g_i|^2$

Consider the SNR in a channel with L order of diversity, $\gamma = \alpha|h|^2$. The random variable $|h|^2$ has a central chi-square distribution with $2L$ degrees of freedom, with $E[|h|^2] = 1$. The PDF of γ is given as [92]:

$$f_\gamma(\gamma) = \frac{\gamma^{L-1}}{(L-1)!\alpha^L} \exp\left(-\frac{\gamma}{\alpha}\right). \tag{4.35}$$

The cumulative density function (cdf), $F_\gamma(\gamma)$, can be obtained by performing integration by parts on (4.35) L times

$$F_\gamma(\gamma) = 1 - \sum_{l=0}^{L} \left(\frac{\gamma}{\alpha}\right)^l \exp\left(-\frac{\gamma}{\alpha}\right). \tag{4.36}$$

This appendix calculates the PDF of $\gamma_i = \alpha|g_i|^2$, formed by ordering M independent realizations of the random variable γ from smallest to largest (i.e. $\gamma_1 \leq \gamma_2 \leq \cdots \leq \gamma_M$). Since the fading channels in all subcarriers are statistically independent, by using order statistics [92], the PDF of γ_i is written as:

$$f_{\gamma_i}(\gamma) = \frac{M!}{i!(M-i-1)!} F_\gamma^i(\gamma)(1 - F_\gamma(\gamma))^{M-i-1} f_\gamma(\gamma). \tag{4.37}$$

For the case of $L = 1$, by substituting (4.35) and (4.36) into (4.37), we get

$$f_{\gamma_i}(\gamma) = \frac{M!}{i!(M-i-1)!\alpha} \sum_{k=0}^{i} \binom{i}{k} (-1)^k \exp\left(-\frac{\gamma(M-i+k)}{\alpha}\right), \tag{4.38}$$

obtained by performing the binomial expansion of $\left[1-\exp\left(-\frac{\gamma}{\alpha}\right)\right]^i$. For the case of $L=2$, after some derivations, we get

$$f_{\gamma_i}(\gamma) = \frac{M!}{i!(M-i-1)!\alpha}\sum_{k=0}^{i}\binom{i}{k}(-1)^k\exp\left(-\frac{\gamma(M-i+k)}{\alpha}\right) \times \sum_{n=0}^{M-i+k-1}\binom{M-i-1+k}{n}\frac{\gamma^{n+1}}{\bar{\gamma}^{n+2}} \tag{4.39}$$

where an additional binomial expansion is performed. Note that $f_{\gamma_i}(\gamma)$ are expressed in terms of the sums of different order terms of $e^{-\gamma}$ and γ to facilitate latter derivations. In a similar manner, we can also derive the PDF for $L>2$.

Appendix 4.C Proof of (4.31)

From (4.23), defining $x=1/\bar{\gamma}$, we get $\mu=(1+x)^{-1/2}$. Clearly, the BER given by (4.23) is m times differentiable at $x=0$, $m=1,2\cdots$, and thus a Maclaurin series with coefficients denoted as a_m corresponding to the mth power can be formed as a power series of x. Note that for $m<L$, $a_j=0$ since substituting $x=0$ into $\frac{d}{dx^m}\left(\frac{1-\mu}{2}\right)^L$ as required in an intermediate step of the Maclaurin expansion gives an answer of zero. Thus, we conclude that $P_L^{fading}(\bar{\gamma})=\sum_{j=L}^{\infty}a_j/\bar{\gamma}^j$.

Also, from direct evaluation, the first few terms are given as $P_{L=1}^{fading}(\bar{\gamma})=\frac{1}{4\bar{\gamma}}-\frac{3}{16\bar{\gamma}^2}+\frac{5}{32\bar{\gamma}^3}+\cdots$ and $P_{L=2}^{fading}(\bar{\gamma})=\frac{3}{16\bar{\gamma}}-\frac{5}{16\bar{\gamma}^2}+\frac{105}{256\bar{\gamma}^3}-\cdots$.

Appendix 4.D Proof of Corollaries

Proof of Corollary 4.1

By definition, $P_{\rm e}(0,\bar{\gamma})\geq P_{\rm low}(0,\bar{\gamma})$. Substituting (4.27) into (4.28) for $i=0$, we get $P_{\rm low}(0,\bar{\gamma})=P_1^{\rm fading}(\bar{\gamma})$, thus concluding the proof that $P_{\rm e}(0,\bar{\gamma})\geq P_1^{\rm fading}(\bar{\gamma})$.

Proof of Corollary 4.2

From (4.30) we get

$$P_{\rm low}(0,\bar{\gamma})=M\sum_{n=0}^{M-1}\binom{M-1}{n}\frac{(n+1)!}{M^{n+2}}P_{n+2}^{\rm fading}(\bar{\gamma}). \tag{4.40}$$

Using (4.31), we see that $P_{\rm low}(0,\bar{\gamma})$ can be asymptotically (as SNR$\rightarrow\infty$) represented by the first term of the summation of (4.40) when $n=0$. The proof is concluded since the first term is $1/M$.

Proof of Corollary 4.3

To prove Corollary 4.3, we first need to prove the following lemma.
Lemma 4.1.

$$\Lambda(j) = \sum_{k=0}^{n} \binom{n}{k} (-1)^k (N-n+k)^{j-1} = 0, \ \forall j = 1, \cdots, n \tag{4.41}$$

Proof. By expanding $(N-n+k)^{j-1}$, $\Lambda(j)$ as defined in (4.41) can be re-written as

$$\Lambda(j) = \sum_{r=0}^{j-1} \binom{j-1}{r} (N-n)^{j-1-r} g(r), \tag{4.42}$$

where $g(r) \triangleq \sum_{k=0}^{n} \binom{n}{k} (-1)^k k^r$. We first consider the range of r when $g(r) = 0$.

Define $f(x) = (x-1)^n = \sum_{k=0}^{n} \binom{n}{k} (-1)^k x^{n-k}$. Differentiating $f(x)$ m times, $m = 0, 1 \cdots, n-1$, and noting the two definitions for $f(x)$, we get

$$\begin{aligned} f^m(x) &= n(n-1)\cdots(n-m+1)(x-1)^{n-m} \\ &= \sum_{k=0}^{n} \binom{n}{k} (-1)^k \Pi_{p=k}^{k+m} (n-p). \end{aligned} \tag{4.43}$$

Letting $x = 1$, and expressing as a power series of k for $(n-k)(n-k-1)\cdots(n-k-m+1) = \sum_{r=0}^{m} c_{j,m} k^j$, where $c_{j,m} \neq 0$, from (4.43) we get

$$f^m(1) = \sum_{k=0}^{n} \binom{n}{k} (-1)^k \sum_{r=0}^{m} c_{j,m} k^j x^{n-k-m} = 0, \tag{4.44}$$

which can be further simplified to

$$f^m(1) = \sum_{r=0}^{m} c_{j,m} g(r) = 0. \tag{4.45}$$

Considering $m = 0$ we get $f(1) = g(0) = 0$. Considering $m = 1$, we get $g(1) = 0$ since $f^1(1) = c_{0,1} g(0) + c_{1,1} g(1) = 0$, where $c_{0,1} = n$ and $c_{1,1} = -1$. Thus, from (4.45), we have by induction $g(m) = 0$ for $m = 0, 1, \cdots, n-1$.

Now, note from (4.42) that for $\Lambda(j) = 0, j = 1, 2, \cdots$, we require that $g(r) = 0$ for $r = 0, \cdots, j-1$. The largest value of j when $\Lambda(j) = 0$ depends on the the maximum range of r such that the expression $g(r) = 0$ is valid. Since we showed earlier that the maximum range of r is when $j = n$, we conclude that $\Lambda(j) = 0$ for $j = 1, 2, \cdots, n$. □

We are now ready to prove Corollary 4.3.

Proof (Corollary 4.3). Note that (i) the diversity advantage of the lower and upper BER bounds are the same since they only differ in SNR by $10\log 10(N)$ and (ii) the diversity advantage of the exact BER is also bounded by that of the lower and upper bounds. Thus, it is determined by that of the lower (or upper) bound, which we need to prove to be at least i. Using (4.31), we can re-write (4.27) as $P_{\mathrm{up}}(i,\bar{\gamma}) = \frac{M!}{i!(M-i-1)!}\sum_{j=1}^{\infty}\frac{a_j}{\bar{\gamma}_{\mathrm{low},i}^{j}}\Lambda(j)$ where

$$\Lambda(j) \triangleq \sum_{k=0}^{i}\binom{i}{k}(-1)^k(M-i+k)^{j-1}. \tag{4.46}$$

From Lemma 4.1, we have $\Lambda(j)=0$ for all $j=1,\cdots,i$, we thus obtain

$$P_{\mathrm{up}}(i,\bar{\gamma}) = \frac{M!}{i!(M-i-1)!}\sum_{j=i+1}^{\infty}\frac{a_j}{\bar{\gamma}_{\mathrm{low},i}^{j}}\Lambda(j). \tag{4.47}$$

As $\bar{\gamma}\to\infty$, $P_{\mathrm{up}}(i,\bar{\gamma})$ approaches a positive constant divided by $\bar{\gamma}_{\mathrm{low},i}^{i+1}$. This implies that the diversity advantage is at least $i+1$ for the ith iteration at high SNR. □

CHAPTER 5

ARQ BY SUBCARRIER ASSIGNMENT

In this chapter[1], we consider two automatic-repeat-request (ARQ) schemes based on subcarrier assignment in OFDM-based systems: single ARQ subcarrier assignment (single ARQ-SA) and multiple ARQ-SA. In single ARQ-SA, data transmitted on a subcarrier in a failed transmission is repeated on a *single assigned* subcarrier in the ARQ transmission. In multiple ARQ-SA, the data is repeated on *multiple assigned* subcarriers in the ARQ transmission. At the receiver, maximum ratio combining is performed on subcarriers that carry the same data. Our goal is to optimize certain system utility functions (such as to minimize bit error rates or to maximize sum capacity) through the choice of the subcarrier assignment. We show that a large class of reasonable system utility functions that we wish to maximize are characterized as Schur-concave. For this class of utility functions, we obtain the optimum subcarrier assignment for single ARQ-SA, and propose a sub-optimum (heuristic) subcarrier assignment scheme for multiple ARQ-SA. Further, to lower the overhead of signaling the subcarrier assignment information, we consider subcarrier grouping methods. Numerical results indicate that substantial throughput improvement can be achieved by appropriate assignments, especially with the use of incremental redundancy at high SNRs.

[1]A large part of this work has been published as "ARQ by Subcarrier Assignment for OFDM-Based Systems" in *IEEE Trans. Signal Process.*, vol. 56, no. 12, pp. 6003–6016, Dec. 2008.

5.1 Introduction

Orthogonal frequency division multiplexing (OFDM) is an effective solution for delivering high data rates over wireless channels with frequency-selective fading. A number of wireless standards such as IEEE 802.11a [9], WiMax [32] and long term 3G evolution [33] have adopted OFDM-based solutions for physical-layer transmission. An OFDM-based system combats multipath fading with the use of a cyclic prefix that, in conjunction with the Fourier transform, converts the frequency selective fading channel into a set of parallel subcarriers experiencing flat fading. It is common to use an automatic repeat request (ARQ) mechanism [7, 8] in OFDM systems when a packet transmission fails. In this mechanism, the transmitter retransmits the data when it fails to receive an acknowledgment (ACK) or receives an explicit negative ACK. We shall study ARQ schemes involving subcarrier assignment in OFDM-based systems.

The system under consideration is a general OFDM system where a linear unitary pre-transform may be applied before the application of the IDFT at the transmitter. Such pre-transformed OFDM (PT-OFDM) systems have been known to offer various advantages such as improved block error rates [88] and reduced transmitter complexity [81]. A PT-OFDM system can also be shown to be equivalent to a system with parallel subcarriers in the frequency domain. In such systems, under the ARQ mechanism, the data in the failed transmission has to be retransmitted over the parallel subcarriers in the event of a packet failure. For clarity, we shall call the failed transmission the original transmission and the associated subcarriers the original subcarriers. The retransmission will be called ARQ transmission and the associated subcarriers will be termed ARQ subcarriers. In this chapter, we consider two ARQ schemes: single ARQ subcarrier assignment (ARQ-SA) scheme and multiple ARQ-SA. In single ARQ-SA, data on an original subcarrier is repeated on a single ARQ subcarrier which may be different from the original subcarrier. We say that the ARQ subcarrier is *assigned* to the original subcarrier. In multiple ARQ-SA, however, zero, one or more ARQ subcarriers may be assigned to an original subcarrier. At the receiver, maximum ratio combining (MRC) is performed on the original subcarrier and all ARQ subcarriers that carry the same data. Subsequently, a single stage of equalization and decoding is carried out.

Our goal is to optimize a certain system metric by choosing the assignment, under the assumption that full channel state information (CSI) is available at both the transmitter and the receiver. We first phrase this optimization problem as one of maximizing a utility function, and show that many utility functions of practical interest that we wish to maximize are Schur-concave. Examples of such utility functions are the sum capacity and the probability of correct reception. Under single ARQ-SA, we show that for Schur-concave utility functions the optimum assignment is to assign the ARQ subcarrier with the strongest SNR to the original subcarrier with the weakest SNR, and so on. That is, if the original subcarriers are ordered with their respective SNRs in decreasing order, then the assigned ARQ subcarriers should be ordered with their respective SNRs in increasing order. Similar results have been obtained recently in the context of relay-assisted communications [93, 94]. To obtain our results, we

employ the theory of majorization [42]; see [95,96] for a recent review and also [97,98] for applications of majorization theory. Under multiple ARQ-SA, however, we find that determining the optimum assignment is an NP-hard problem. Hence, we propose a heuristic scheme.

The ARQ-SA schemes we propose here are aimed at exploiting CSI effectively, so as to realize the maximum gain offered through an appropriate assignment. Fixed assignments that do not exploit CSI have been considered in the literature [39–41]. In these works, a diversity effect is realized by retransmitting data on an ARQ subcarrier with a channel that is independent from that of the original subcarrier. To this end, *cyclic assignments* are used, where ARQ subcarriers are cyclic-shifted versions of the original subcarriers. If the cyclic shift is larger than the coherence bandwidth, then data is transmitted on subcarriers that experience independent channel fades, even in quasi-static channels. This clearly offers an improvement over a simple scheme where no assignment is done. However, in cyclic assignments, for some channel realizations data on a weak original subcarrier may again be retransmitted on a weak ARQ subcarrier, resulting in poor performance. We address this shortcoming by using the available CSI to develop better assignment strategies. In [99], CSI was also employed to develop an ARQ scheme based on Chase combining. Although [99] considered the problem of choosing the *group* of ARQ subcarriers for retransmission (specifically those with SNRs above a certain threshold), the problem of how to exactly perform the assignment was not addressed.

The chapter is organized as follows. In Section 5.2, we describe the general OFDM system with ARQ-SA and consider a number of utility functions that characterize system performance. The assignment problems for single ARQ-SA and multiple ARQ-SA are formulated in Section 5.3. Algorithms for subcarrier assignment that solve these problems and their complexity are presented in Section 5.4. The optimality of these algorithms is investigated in Section 5.5. The overhead of signaling the subcarrier assignment is analyzed in Section 5.6, where subcarrier grouping techniques for reducing this overhead are presented. Section 5.7 presents throughput analysis for the subcarrier assignment schemes. In Section 5.8, simulation results are presented to test the efficacy of our algorithms. Conclusions are drawn in Section 5.9.

Notation: We use bold lower case letters to denote column vectors and bold upper case letters to denote matrices. The superscripts *, ^{T} and ^{H} denote complex conjugate, transpose and Hermitian, respectively. The (m,n)th element of matrix $\mathbf{W}$ is denoted by w_{mn} and the mth element of vector $\mathbf{w}$ is denoted w_m. The identity matrix is denoted as $\mathbf{I}$ while the all-ones vector is denoted as $\mathbf{1}$.

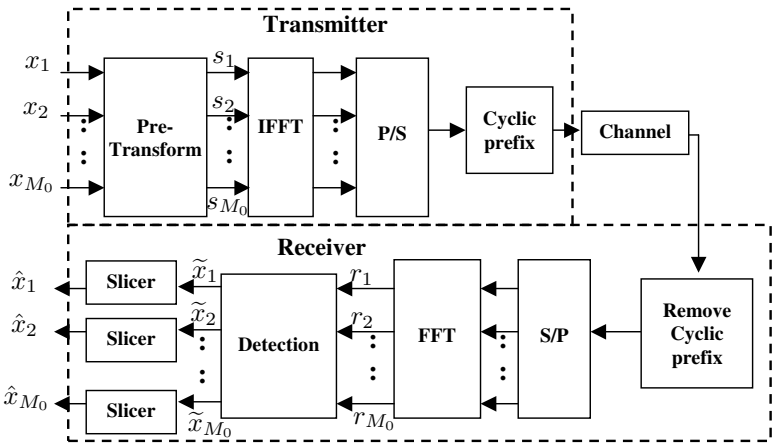


Fig. 5.1: An OFDM system with M_0 subcarriers.

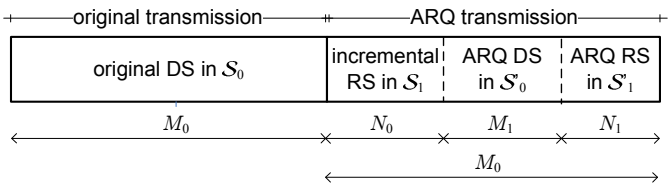
5.2 System Description

5.2.1 OFDM Systems

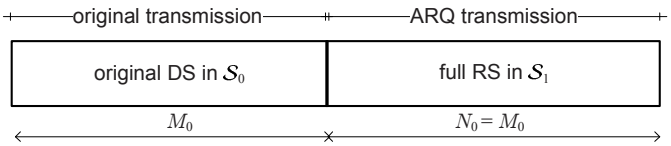
In Fig. 5.1 we show a general OFDM system including a possible pre-transform. We consider a system with M_0 symbols, each of unit power, represented as $\mathbf{x} = [x_1, x_2, \ldots, x_{M_0}]^T$. We shall see that each symbol x_m may either carry data or redundancy that is used to improve the probability of detecting previously failed transmissions. The vector $\mathbf{x}$ is linearly transformed into M_0 subcarriers in the frequency domain as $\mathbf{s} = [s_1, s_2, \ldots, s_{M_0}]^T = \mathbf{W}\mathbf{x}$, where $\mathbf{W}$ is an $M_0 \times M_0$ transformation matrix. For simplicity, we consider either an OFDM system where $\mathbf{W}$ is an identity matrix or the *pre-transformed OFDM* (PT-OFDM) system where $\mathbf{W}$ is unitary with constant-amplitude entries [100]. These choices are prevalent in current wireless systems [9, 32, 33]. The block of modulation symbols $\mathbf{s}$ is then passed through an inverse discrete Fourier transform, usually implemented using the inverse fast Fourier transform (IFFT). After performing a parallel-to-serial (P/S) conversion (its inverse operation is denoted as S/P), we insert a cyclic prefix with duration not shorter than the maximum channel delay spread so as to avoid inter-OFDM symbol interference. Finally, the PT-OFDM symbol is transmitted. At the receiver, the samples of the received signal corresponding to the cyclic prefix are removed. After FFT, the received subcarrier vector in the frequency domain $\mathbf{r} = [r_1, r_2, \ldots, r_{M_0}]^T$ can be expressed as

$$\mathbf{r} = \mathbf{\Gamma}\mathbf{s} + \mathbf{v} = \mathbf{W}\mathbf{x} + \mathbf{v}, \tag{5.1}$$

where $\mathbf{v} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$ is i.i.d. circularly symmetric complex AWGN with zero mean and unit variance, while $\mathbf{\Gamma} = \mathrm{diag}(h_1, h_2, \ldots, h_{M_0})$ is a diagonal matrix. Here, $h_m = \sum_l \tilde{h}_l \exp(-\mathrm{j}2\pi lm/M_0)$, $m = 1, \cdots, M_0$, is the mth channel response coefficient in the frequency domain, assuming a sample-spaced L_hth order finite-impulse response channel model with coefficients $\{\tilde{h}_l, l = 0, \cdots, L_h\}$.



(a) In the ARQ transmission, incremental redundancy is sent to protect the original DSs. Additional data is sent as ARQ DSs, and additional redundancy to protect the ARQ DSs is sent as ARQ RSs.



(b) In the ARQ transmission, full redundancy is sent to protect the original DSs. No additional data is sent.

Fig. 5.2: Transmission structure for the original and the first ARQ transmission. Redundancy for the original data symbols (DSs) is sent generally by using (a) incremental redundancy symbols (RSs), or as a special case, by using (b) full RSs.

5.2.2 Transmission Scheme

We define an *ARQ round* to consist of an original transmission and the subsequent ARQ transmissions, before the next original transmission begins. The ARQ round ends if all the data sent so far has been recovered, or if a maximum number of ARQ transmissions has been reached. After the ARQ round has ended, all past transmissions are discarded from memory. To initiate another ARQ round, an independent original transmission is sent. Hence, the transmissions within any ARQ round are *independent* of those in other ARQ rounds. Fig. 5.2 shows the transmission structure of an ARQ round consisting of the original transmission and the first ARQ transmission. We assume that each transmission uses one OFDM symbol with subcarriers in the set $\mathcal{S} = \{1, \cdots, M_0\}$; extensions to multiple OFDM symbols are straightforward and are not treated in this chapter.

For exposition, let us first focus on Fig. 5.2(a). Here, subcarriers used for a common purpose are grouped for clarity, but they need not be neighboring subcarriers. In general, each symbol x_i in (5.1) is used for transmission either as a data symbol (DS) or as a redundancy symbol (RS). Specifically, in the original transmission, *original DSs* comprising of $\{x_i, i \in \mathcal{S}_0\}$ are used to send data, where $\mathcal{S}_0$ is the set of subcarriers used. Clearly, all subcarriers should be used for transmission, hence $\mathcal{S}_0 = \mathcal{S}$ and the size of the set is $|\mathcal{S}_0| = M_0$. When at least one bit error occurs in the original transmission, the ARQ transmission is triggered, for example by feeding back a negative ACK (NACK) to the transmitter. In the ARQ transmission, RSs comprising of $\{x_i, i \in \mathcal{S}_1\}$ with size $|\mathcal{S}_1| = N_0$ are sent as redundancy for the original DSs. We refer to these RSs generally as *incremental RSs*. If $\mathcal{S}_0 = \mathcal{S}_1$ and thus $N_0 = M_0$, we refer to these

RSs specifically as *full RSs*, and Fig. 5.2(a) becomes Fig. 5.2(b) as a special case. In general, the remaining $M_0 - N_0$ subcarriers in the ARQ transmission are then split into two disjoint sets $\mathcal{S}_0'$ of size M_1 and $\mathcal{S}_1'$ of size M_1. The set $\mathcal{S}_0'$ carries *ARQ DSs* that are used to send more data. The set $\mathcal{S}_1'$ carries *ARQ RSs* that are used as redundancy for the ARQ DSs. Clearly, $M_0 = N_0 + M_1 + N_1$.

If the original DSs or ARQ DSs are still not recovered after the first ARQ transmission is sent, a *second* ARQ transmission consisting of more RSs and DSs can be sent, by a straightforward generalization of Fig. 5.2(a). For clarity of presentation, henceforth we allow at most one ARQ transmission to be sent, as depicted in Fig. 5.2(a). With this restriction, simulation results in Section 5.8 show that substantial performance can already be achieved; better performance would be achieved with more ARQ transmissions.

In this chapter, our use of incremental redundancy is more general (with arbitrary N_0, M_1, N_1), with full redundancy as a special case (with $N_0 = M_0, M_1 = N_1 = 0$). The ARQ schemes in the literature typically employ only full redundancy, e.g. [39–41, 70], although the redundancy is referred to as incremental redundancy.

5.2.3 Incremental RSs for Original Data Symbols

We consider how to assign incremental RSs (and full RSs as a special case) to original DSs. Recall that $\mathcal{S}_0$ and $\mathcal{S}_1$ are the sets of subcarrier indices used by the original DSs and incremental RSs, respectively, where $\mathcal{S}_0 = \mathcal{S}$ and $\mathcal{S}_1 \subseteq \mathcal{S}$. The channel coefficients in the frequency domain in the original and ARQ transmissions are denoted as $h_m, m \in \mathcal{S}_0$, and as $g_n, n \in \mathcal{S}_1$, respectively. We call h_m the mth original subcarrier and g_n the nth ARQ subcarrier. For a time-invariant channel, assuming full redundancy is used, we have $h_m = g_n$ for $m = n$. This scenario is commonly considered in the literature and is covered in our formulation. We denote the power of the original subcarrier and ARQ subcarrier by $\alpha_m = |h_m|^2$ and $\beta_n = |g_n|^2$, respectively. Since the noise variance is set to one, α_m and β_n are also the SNRs of the original and ARQ subcarrier, respectively.

The received signal (5.1) can be expressed on a per-subcarrier basis as

$$r_m = h_m \mathbf{w}_m^H \mathbf{x} + v_m,\ m \in \mathcal{S}_0, \tag{5.2}$$

where $\mathbf{w}_m^H$ is the mth row of the transform $\mathbf{W}$. Suppose that at least one bit in $\mathbf{x}$ is not received correctly and ARQ is triggered. The signal $\mathbf{w}_m^H \mathbf{x}$ carried by original subcarrier m is then repeated in the *assigned* ARQ subcarriers in the ARQ transmission. The set of indices of the ARQ subcarriers assigned to original subcarrier m are denoted as $\mathcal{A}(m) \subseteq \mathcal{S}_1$. These ARQ subcarriers are received as

$$r_n' = g_n \mathbf{w}_m^H \mathbf{x} + v_n',\ n \in \mathcal{A}(m), \tag{5.3}$$

where $v_n' \sim \mathcal{CN}(0, 1)$. To detect $\mathbf{x}$, we employ MRC on all the received signals that

carry $\mathbf{w}_m^H \mathbf{x}$ to give

$$\widetilde{r}_m = h_m^* r_m + \sum_{n \in \mathcal{A}(m)} g_n^* r_n', \ m \in \mathcal{S}_0. \tag{5.4}$$

Since the noise in all received signals is independent, it follows that the *effective SNR* of $\widetilde{r}_m$ in the frequency domain is given by summing the SNRs of the original subcarrier and the assigned ARQ subcarriers, which gives

$$\gamma_m = \alpha_m + \sum_{n \in \mathcal{A}(m)} \beta_n, \ m \in \mathcal{S}_0. \tag{5.5}$$

Besides the assignment carried out via $\mathcal{A}(m)$, we note that the effective SNR depends also on the subcarrier sets $\mathcal{S}_0$ and $\mathcal{S}_1$ used for (re)transmissions.

We detect $\mathbf{x}$ based on $\widetilde{r}_m, m \in \mathcal{S}_0$, at the receiver. Various detectors can be used, such as the maximum likelihood detector (MLD), iterative detectors or linear-equalizer based detectors. For linear-equalizer based detectors, the signal before the slicers is denoted as $\widetilde{\mathbf{x}}$, see Fig. 5.1. It can be expressed as $\widetilde{\mathbf{x}} = \mathbf{G}\mathbf{r}$, where $\mathbf{G}$ is a linear equalizer. It is common to use the zero-forcing (ZF) or minimum mean-square-error (MMSE) equalizer, given respectively by

$$\mathbf{G}_{\text{ZF}} = \mathbf{\Omega}^{-1}, \tag{5.6a}$$

$$\mathbf{G}_{\text{MMSE}} = (\mathbf{\Omega}^H \mathbf{\Omega} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{\Omega}^H, \tag{5.6b}$$

where $\mathbf{\Omega} \triangleq \text{diag}(\sqrt{\gamma_1}, \cdots, \sqrt{\gamma_{M_0}})\mathbf{W}$.

5.2.4 Redundancy for ARQ Data Symbols

In the previous section, we saw that incremental RSs are assigned to original DSs via the assignment $\mathcal{A}(m)$. The assignment of ARQ RSs to ARQ DSs is carried out in essentially the same way, but with the "retransmission" always triggered. In particular, (5.2)–(5.5) apply for the case of assigning ARQ RSs to ARQ DSs by replacing $\mathcal{S}_0$ with $\mathcal{S}_0'$ and $\mathcal{S}_1$ with $\mathcal{S}_1'$, and a new assignment in place of $\mathcal{A}(m)$ is used. Henceforth, it suffices to consider the problem of assigning incremental RSs to original DSs as considered in Section 5.2.3, since the problem of assigning ARQ RSs to ARQ DSs is similar.

Although we consider at most one ARQ transmission, (5.2)–(5.5) can be easily generalized to any arbitrary number of ARQ transmissions, so that the subcarrier assignment problem remains essentially unchanged. Details are provided in Appendix 5.A.

5.2.5 Utility Functions

We describe several utility functions $\phi(\boldsymbol{\gamma})$ commonly used to reflect system performance, as a function of the effective-SNR vector $\boldsymbol{\gamma}$ with elements γ_m from (5.5). We

seek to maximize these utility functions by appropriately choosing the subcarrier assignment. To illuminate this problem, we define the utility functions with respect to the original DSs. The problem remains essentially the same if we instead maximize the utility with respect to the ARQ DSs. This is because to reflect the new utility functions, we only need to replace $\mathcal{S}_0$ with $\mathcal{S}_0'$, and $\mathcal{S}_1$ with $\mathcal{S}_1'$.

5.2.5.1 OFDM and PT-OFDM

For OFDM and PT-OFDM systems, we consider these utility functions:

$$\phi_{\min}(\boldsymbol{\gamma}) = \min\{\gamma_1, \cdots, \gamma_M\}, \tag{5.7a}$$

$$\phi_{\mathrm{MI}}(\boldsymbol{\gamma}) = \sum_{m=1}^{M} \log(1+\gamma_m). \tag{5.7b}$$

In (5.7a), $\phi_{\min}$ is the minimum of the effective SNR over all subcarriers. The uncoded symbol error performance is often dominated by weak subcarriers experiencing deep fades. To reduce the effects of fading, the effective SNR should be made as flat as possible across the subcarriers; one way to do this is to maximize $\phi_{\min}$. To give an intuitive explanation of our subsequent results, we will make frequent use of $\phi_{\min}$. In (5.7b), ϕ_{MI} is the sum of the mutual information between x_m and $\widetilde{r}_m$ in (5.4) after MRC is performed, assuming that $\mathbf{x}$ is i.i.d. Gaussian distributed. We note that ϕ_{MI} indicates the number of bits that can be reliably transmitted with a Gaussian codebook, if ideal channel coding is carried out.

5.2.5.2 PT-OFDM

For PT-OFDM systems, to simplify implementations, we may use either the ZF or MMSE equalizer (5.6). Let $\widetilde{x}_m$ be the equalized signal before slicing. The SNR of $\widetilde{x}_m$ after ZF equalization and the signal-to-interference noise ratio (SINR) after MMSE equalization is denoted as $\phi_{\mathrm{PT-ZF}}$ and $\phi_{\mathrm{PT-MMSE}}$, respectively. Both are appropriate measures to maximize since error probabilities typically decrease as SNR or SINR increases. When $\mathbf{W}$ is unitary with constant-amplitude entries, we obtain (see for example [100]) the SNR and SINR for subcarrier m as

$$\phi_{\mathrm{PT-ZF}}(\boldsymbol{\gamma}) = \frac{M}{\sum_{m=1}^{M} \gamma_m^{-1}}, \tag{5.7c}$$

$$\phi_{\mathrm{PT-MMSE}}(\boldsymbol{\gamma}) = \frac{1}{\delta^{-1}-1}, \quad \delta \triangleq \frac{1}{M}\sum_{m=1}^{M} \frac{1}{1+\gamma_m^{-1}}, \tag{5.7d}$$

respectively. We note that the SNR or SINR is independent of m.

We refer to all the original DSs and their redundancy symbols, or all the ARQ DSs and their redundancy symbols, as a *block*. The block error rate (BLER) is defined as the probability that at least one bit error occurs in a block. This is appropriate if each

block uses a separate error detection code, and a block is discarded when any bit error occurs. Two important measures of performance are the BER and block error rate (BLER). We use QPSK modulation throughout this chapter. For ZF equalization, the noise after equalization is Gaussian distributed, and the BER (on any subcarrier) is thus given by $P_{\mathrm{e}} = Q\left(\sqrt{\phi_{\mathrm{PT-ZF}}/2}\right)$, where $Q(x) = \int_x^\infty \exp(-y^2/2)/\sqrt{2\pi}\mathrm{d}y$. The BLER is given by $1-(1-P_{\mathrm{e}})^{2M}$, since there are $2M$ bits in an OFDM symbol with QPSK modulation. We see that both the BER and BLER indeed decrease monotonically as $\phi_{\mathrm{PT-ZF}}$ increases.

5.2.5.3 OFDM

For OFDM systems, the effective SNR γ_m for subcarrier m remains the same after ZF or MMSE equalization, since these equalizations involves only a scalar multiplication. An appropriate measure to *minimize* is the expected BER, $P_{\mathrm{e}}(\gamma_m)$, summed over all m. The utility function to *maximize* is then the negative of this measure. Hence, the utility function is

$$\phi_{\mathrm{OFDM-BER}}(\boldsymbol{\gamma}) = -\sum_{m=1}^{M} Q\left(\sqrt{\gamma_m/2}\right). \tag{5.7e}$$

We can alternatively minimize the BLER. This is equivalent to *maximizing* the probability that all the bits in the block are successfully detected, given by

$$\phi_{\mathrm{OFDM-BLER}}(\boldsymbol{\gamma}) = \prod_{m=1}^{M} \left(1 - Q\left(\sqrt{\gamma_m/2}\right)\right). \tag{5.7f}$$

5.3 Problem Formulation

We consider the problem of finding the optimal subcarrier assignment for a *given* choice of subcarrier sets $\mathcal{S}_1, \mathcal{S}_0', \mathcal{S}_1'$ (this choice also fixes the parameters N_0, M_1, N_1). In practice, the choice of these subcarrier sets can be predetermined to optimize system performance for a given average SNR, while the subcarrier assignment is optimized during run time whenever small-scale fading leads to changes in the channel. Since our problem formulation holds for arbitrary $\mathcal{S}_1, \mathcal{S}_0', \mathcal{S}_1'$, we defer the choice of the subcarrier sets to Section 5.8. Briefly, we have chosen the subcarrier sets such that, as much as possible, each re-transmitted symbol experiences an uncorrelated channel, while N_0, M_1, N_1 are optimized via simulations.

We now describe two ARQ-SA schemes, namely single ARQ-SA and multiple ARQ-SA.

We assume that the CSI in the form of the SNRs of the original and ARQ subcarriers is known to the transmitter. The CSI can be estimated at the transmitter when the channel is reciprocal, such as in IEEE 802.11a system [9], or by an explicit CSI

feedback, which is used in long term 3G evolution systems [33]. In practice, if the time duration between an original transmission and its ARQ transmissions is less than the channel coherence time, it is reasonable to assume that the CSI is known.

5.3.1 ARQ Subcarrier Assignment (ARQ-SA)

To simplify the description, we define $a_{mn} = 1$ if ARQ subcarrier n is assigned to original subcarrier m, i.e., if $n \in \mathcal{A}(m)$; we define $a_{mn} = 0$ otherwise. The ARQ-SA is completely described by a $M_0 \times N_0$ matrix $\mathbf{A}$ with binary entries a_{mn}. Hence, the effective SNR (5.5) can be written in vector form as

$$\boldsymbol{\gamma}(\mathbf{A}) = \boldsymbol{\alpha} + \mathbf{A}\boldsymbol{\beta}, \tag{5.8}$$

where $\boldsymbol{\beta}, \boldsymbol{\alpha}$ are the vectors of β_m, α_m, respectively. In (5.8), we emphasize the dependence of $\boldsymbol{\gamma}$ on $\mathbf{A}$ explicitly.

5.3.2 ARQ-SA Schemes

For maximum performance gain, clearly each ARQ subcarrier should be assigned to at least one original subcarrier. To employ MRC directly, an ARQ subcarrier cannot be assigned to two or more original subcarriers; otherwise, more advanced techniques, such as MLD or interference cancelation, would be required for detection. Hence, to keep the receiver processing simple, in all our schemes each ARQ subcarrier is assigned to exactly one original subcarrier. Since $\mathbf{A}$ is a binary matrix, this constraint is equivalent to imposing the condition $\sum_{m \in \mathcal{S}_0} a_{mn} = 1,\ n \in \mathcal{S}_1$, i.e.,

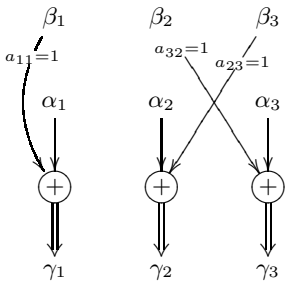
$$\mathbf{A}^T \mathbf{1} = \mathbf{1}. \tag{5.9}$$

Two ARQ-SA schemes are possible depending on whether multiple ARQ subcarriers can be assigned to an original subcarrier.

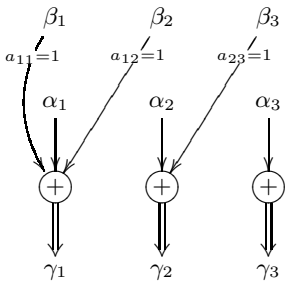
5.3.2.1 Single ARQ-SA

In single ARQ-SA, we impose the condition that each original subcarrier is either assigned to one ARQ subcarrier, or not assigned at all. That is, we do not allow multiple ARQ subcarriers to be assigned to any original subcarriers. We shall see in Section 5.5 that this condition simplifies an otherwise NP-hard problem to a problem that has an optimal solution with polynomial-order complexity.

Together with constraint (5.9), the imposed condition under single ARQ-SA implies that we must have $N_0 \leq M_0$. For convenience of description, henceforth we add $M_0 - N_0$ *virtual subcarriers* with zero SNRs to the set of ARQ subcarriers $\mathcal{S}_1$. Note that an original subcarrier is *not* physically assigned to any ARQ subcarrier if the ARQ subcarrier turns out to be a virtual subcarrier. Equivalently, we pad $\boldsymbol{\beta}$ with zeros so that its length becomes M_0, and so $\mathbf{A}$ becomes a $M_0 \times M_0$ square matrix.



(a) Only one ARQ subcarrier is assigned to each original subcarrier.



(b) Multiple ARQ subcarriers can be assigned to each original subcarrier.

Fig. 5.3: An ARQ subcarrier n is assigned to an original subcarrier m if $a_{mn} = 1$. The effective SNR γ_m is the sum of the SNR of the original subcarrier α_m and all SNRs of the assigned ARQ subcarriers β_n where $a_{mn} = 1$.

By including virtual subcarriers, the imposed condition is equivalent to setting the row sums of $\mathbf{A}$ to one, i.e.,

$$\mathbf{A1} = \mathbf{1}. \tag{5.10}$$

From (5.9) and (5.10), $\mathbf{A}$ is therefore a permutation matrix, and single ARQ-SA reduces to finding an optimal permutation (not necessarily unique) that maximizes the utility function.

As an example, consider Fig. 5.3(a). We see that each ARQ subcarrier, say with SNR β_n, is assigned to one original subcarrier, say with SNR α_m, to give an effective SNR of $\gamma_m = \alpha_m + \beta_n$.

Under single ARQ-SA, the optimization problem becomes:

Problem Single ARQ-SA:
Find $M_0 \times M_0$ $\mathbf{A}$ that solves

$$\begin{aligned} &\text{maximize } \phi(\boldsymbol{\gamma}), \text{ where } \boldsymbol{\gamma} = \boldsymbol{\alpha} + \mathbf{A}\boldsymbol{\beta}, \\ &\text{subject to } \mathbf{A1} = \mathbf{1}, \\ &\qquad\qquad \mathbf{A}^T\mathbf{1} = \mathbf{1}, \\ &\qquad\qquad a_{mn} \in \{0,1\}, \ \forall m,n. \end{aligned} \tag{5.11}$$

5.3.2.2 Multiple ARQ-SA

If an original subcarrier is already very strong, with high probability the data would be recovered. Hence, we should not assign any ARQ subcarrier to it. Instead, we

could assign multiple ARQ subcarriers to boost the performance of an original subcarrier that is very weak. To improve system performance and to provide a more general framework, in the multiple ARQ-SA scheme we allow zero, one or more ARQ subcarriers to be assigned to an original subcarrier. To this end, we remove the constraint (5.10) under multiple ARQ-SA. Thus, multiple ARQ-SA applies for any N_0 and M_0, unlike for single ARQ-SA which can be used only if $N_0 \leq M_0$.

An example is given in Fig. 5.3(b) where

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}. \tag{5.12}$$

In this case, two, one, and no ARQ subcarriers are assigned to the first, second, and third original subcarriers, respectively.

Without the constraint of (5.10), under multiple ARQ-SA the optimization problem becomes:

Problem Multiple ARQ-SA:
Find $M_0 \times N_0$ $\mathbf{A}$ that solves

$$\begin{aligned} &\text{maximize } \phi(\boldsymbol{\gamma}), \text{ where } \boldsymbol{\gamma} = \boldsymbol{\alpha} + \mathbf{A}\boldsymbol{\beta}, \\ &\text{subject to } \mathbf{A}^T\mathbf{1} = \mathbf{1}, \\ &\qquad a_{mn} \in \{0,1\}, \ \forall m,n. \end{aligned} \tag{5.13}$$

5.4 Algorithms for ARQ-SA schemes

We begin by providing Algorithm 5.1 and Algorithm 5.2 to solve Problem Single ARQ-SA (5.11) and Problem Multiple ARQ-SA (5.13), respectively, and then discuss their complexity. Algorithm 5.1 is optimal, while Algorithm 5.2 is sub-optimal; a detailed discussion on their optimality will be given in the Section 5.5.

5.4.1 Algorithm 5.1 for Problem Single ARQ-SA

In Algorithm 5.1, we order the original subcarriers increasingly and the ARQ subcarriers decreasingly according to their SNRs. Then, we assign the mth strongest ARQ subcarrier to the mth weakest original subcarrier for all m. The ordering among subcarriers with the same value is arbitrary (this does not change the effective SNR nor the utility function). By pairing strong ARQ subcarriers with weak original subcarriers, Algorithm 5.1 produces effective SNRs that do not fluctuate significantly across subcarriers. Consequently, we expect that the minimum effective SNR $\phi_{\min}$ is increased as compared to random pairings of ARQ and original subcarriers. Algorithm 5.1 is given as follows.

Algorithm 5.1 For solving Problem Single ARQ-SA.

Initialization with inputs $\boldsymbol{\alpha}, \boldsymbol{\beta}$:

- set $\mathbf{A} = \mathbf{0}$;
- order $\boldsymbol{\beta}$ decreasingly to obtain $\boldsymbol{\beta}_{\downarrow}$, so that $\beta_{n(1)} \geq \cdots \geq \beta_{n(N_0)}$, where $n(l)$ is the ordered index;
- order $\boldsymbol{\alpha}$ increasingly to obtain $\boldsymbol{\alpha}_{\uparrow}$, so that $\alpha_{m(1)} \leq \cdots \leq \alpha_{m(M_0)}$, where $m(l)$ is the ordered index.

Iteration $l = 1, 2, \cdots, N_0$:

- assign ARQ subcarrier $n(l)$ to original subcarrier $m(l)$, i.e., set $a_{m(l),n(l)} = 1$.

5.4.2 Algorithm 5.2 for Problem Multiple ARQ-SA

We iteratively assign ARQ subcarriers to the original subcarriers, subcarrier by subcarrier. For initialization, we set the effective SNR as the SNR of the original subcarriers. In each iteration, the strongest ARQ subcarrier that has not been assigned so far is assigned to the original subcarrier with the smallest effective SNR. After assignment, the effective SNR is updated in each iteration to include the contribution from the additional ARQ subcarrier. Notice that Algorithm 5.2 imitates Algorithm 5.1 so as to maximize the effective SNR in a greedy manner. In Algorithm 5.2, we explicitly allow multiple ARQ subcarriers to be assigned to an original subcarrier, otherwise Algorithms 5.1 and 5.2 are clearly equivalent.

Algorithm 5.2 For solving Problem Multiple ARQ-SA.

Initialization with inputs $\boldsymbol{\alpha}, \boldsymbol{\beta}$:

- set $\mathbf{A} = \mathbf{0}$ and $\boldsymbol{\gamma} = \boldsymbol{\alpha}$;
- order $\boldsymbol{\beta}$ decreasingly to obtain $\boldsymbol{\beta}_{\downarrow}$, so that $\beta_{n(1)} \geq \cdots \geq \beta_{n(N_0)}$, where $n(l)$ is the ordered index.

Iteration $l = 1, 2, \cdots, N_0$:

- find smallest effective SNR in $\boldsymbol{\gamma}$ and denote its index as $m(l)$;
- assign ARQ subcarrier $n(l)$ to original subcarrier $m(l)$, i.e., set $a_{m(l),n(l)} = 1$;
- update the effective channel power as:
  $\gamma_{m(l)} = \gamma_{m(l)} + \beta_{n(l)}$.

5.4.3 Complexity

For the cyclic assignment considered in the literature [39–41], practically no complexity is required in determining the assignment during run time, since it is a fixed assignment independent of the CSI. In Algorithms 5.1, 5.2, the assignment has to be re-computed during run time, whenever the channel changes. It is thus important to consider this assignment complexity. To this end, we let $M_0 = N_0 = M$ and consider

how the complexity scales with M. We note that the complexity of ordering or sorting M items is $O(M \log M)$ by using, for example, `merge sort` [101].

5.4.3.1 Algorithm 5.1

The initialization requires two sorting operations for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The actual assignments are linear in complexity, involving a simple recording of the assignment solution in memory. Hence, Algorithm 5.1 has a complexity of $O(M \log M)$.

5.4.3.2 Algorithm 5.2

The initialization requires one sorting operation for $\boldsymbol{\beta}$. Suppose that in the initialization we also sort $\boldsymbol{\gamma}$ (equals $\boldsymbol{\alpha}$). We now consider the complexity of the first step of iteration l in Algorithm 5.2, namely, to find the smallest effective SNR in $\boldsymbol{\gamma}$; the remaining steps of iteration l are less complex and incur only a linear complexity. For $l = 1$, $\boldsymbol{\gamma}$ has already been sorted, so the weakest subcarrier can be found immediately. For $l = 2, \cdots, M$, $\boldsymbol{\gamma}$ has already been sorted *except* for the effective SNR $\gamma_{m(l)}$ that has just been updated in the previous iteration (corresponding to the third step of iteration $l-1$ in Algorithm 5.2). To re-sort $\boldsymbol{\gamma}$, we only need to remove $\gamma_{m(l)}$ and appropriately insert[2] it back to $\boldsymbol{\gamma}$. Since the vector $\boldsymbol{\gamma}$ is already sorted if we exclude $\gamma_{m(l)}$, at most $M-1$ comparisons are required to find the appropriate place to insert $\gamma_{m(l)}$. This insertion is similarly carried out in the sorting algorithm `insertion sort` [101], where the worst-case complexity for each insertion is given by $O(M)$, even after taking into account the number of comparisons and shifts required to adjust the storage of the output. Since there are $M-1$ iterations that require re-sorting, a total complexity of $O(M^2)$ is required. As the sortings in the initialization require a smaller complexity of $O(M \log M)$, we conclude that Algorithm 5.2 has an overall complexity of $O(M^2)$.

5.5 Optimality of Proposed Algorithms

In this section, we show that for utility functions (5.7), Algorithm 5.1 solves Problem Single ARQ-SA (5.11) optimally. We also make comments on the sub-optimality of Algorithm 5.2 for solving Problem Multiple ARQ-SA (5.13).

5.5.1 Algorithm 5.1 for Problem Single ARQ-SA

In order to prove optimality, we need a few results from the theory of majorization [42]. We first introduce the notions of majorization and Schur-concavity.

[2] In implementation, this insertion operation is carried out by updating a list of pointers that tracks the orderings.

Definition of Majorization: For any $\mathbf{x}, \mathbf{y} \in \Re^M$, we say that $\mathbf{x}$ is *majorized* by $\mathbf{y}$ (or $\mathbf{y}$ majorizes $\mathbf{x}$), denoted as $\mathbf{x} \prec \mathbf{y}$, if

$$\sum_{m=1}^{k} x_{[m]} \leq \sum_{m=1}^{k} y_{[m]}, \qquad 1 \leq k \leq M-1, \tag{5.14a}$$

$$\sum_{m=1}^{M} x_m = \sum_{m=1}^{M} y_m. \tag{5.14b}$$

Here, the subscript $[m]$ denotes a decreasing ordering such that $x_{[1]} \geq x_{[2]} \geq \cdots \geq x_{[M]}$.

Definition of Schur-concave functions: A real-valued function ϕ defined on a set $\mathcal{A} \subseteq \Re^M$ is said to be *Schur-concave on* $\mathcal{A}$ if

$$\mathbf{x} \prec \mathbf{y} \text{ on } \mathcal{A} \Rightarrow \phi(\mathbf{x}) \geq \phi(\mathbf{y}). \tag{5.15}$$

Thus, a Schur-concave function ϕ defined on vectors from a set $\mathcal{A}$ achieves its maximum at a vector that is majorized by all other vectors in the set. To test for Schur-concavity, the following two lemmas in [42] can be used. By definition, a function is symmetric if it is invariant under permutation of its variables.

Lemma 5.1 ([42, Chap. 3.A.4])**.** *Let $\mathcal{A} \subset \Re^M$ be a symmetric, non-empty convex set and let $\phi : \mathcal{A} \to \Re$ be continuously differentiable. The function ϕ is Schur-concave on $\mathcal{A}$ if and only if*

$$\phi \text{ is symmetric on } \mathcal{A}, \qquad \text{and} \tag{5.16a}$$

$$(x_i - x_j)\left(\phi_i'(\mathbf{x}) - \phi_j'(\mathbf{x})\right) \leq 0 \text{ for all } \mathbf{x} \in \mathcal{A}, \tag{5.16b}$$

where $\phi_i'(\mathbf{x}) = \partial\phi(\mathbf{x})/\partial x_i$ denotes the partial derivative of ϕ with respect to x_i.

Lemma 5.2 ([42, Chap. 3.C.2])**.** *If ϕ is symmetric and concave on $\mathcal{A}$, then ϕ is Schur-concave on $\mathcal{A}$.*

Without loss of generality, we can assume that the original subcarriers are ordered increasingly based on their SNRs. Our main result can then be summarized as follows.

Theorem 5.1. *Let $\boldsymbol{\alpha}_\uparrow$ be increasingly ordered such that $\alpha_1 \leq \cdots \leq \alpha_{M_0}$ and $\boldsymbol{\beta}_\downarrow$ be decreasingly ordered such that $\beta_1 \geq \cdots \geq \beta_{M_0}$. For a Schur-concave function ϕ,*

$$\phi(\boldsymbol{\alpha}_\uparrow + \boldsymbol{\beta}_\downarrow) \geq \phi(\boldsymbol{\alpha}_\uparrow + \boldsymbol{\beta}_\pi) \tag{5.17}$$

for any $\boldsymbol{\beta}_\pi$ which is a permutation of $\boldsymbol{\beta}$. Alternatively, we have $\arg\max_{\boldsymbol{\beta}_\pi} \phi(\boldsymbol{\alpha}_\uparrow + \boldsymbol{\beta}_\pi) = \boldsymbol{\beta}_\downarrow$.

Proof. From [42, Chap. 6.A.2], we have $\boldsymbol{\alpha}_\uparrow + \boldsymbol{\beta}_\downarrow \prec \boldsymbol{\alpha}_\uparrow + \boldsymbol{\beta}_\pi$ for any $\boldsymbol{\beta}_\pi$ which is a permutation of $\boldsymbol{\beta}$. It follows from the definition of Schur-concavity that (5.17) holds. □

Theorem 5.1 shows that, for a Schur-concave function ϕ, the optimal single ARQ-SA is to assign decreasingly-ordered ARQ subcarriers to the original subcarriers, which is equivalent to Algorithm 5.1. Theorem 5.2 particularizes this result to the utility functions (5.7) which we have considered in this chapter.

Theorem 5.2. *Algorithm 5.1 optimally solves ARQ-SA for the utility functions (5.7).*

Proof. By using Theorem 5.1, it is sufficient to show that the functions (5.7) are Schur-concave. We note that (5.7) are all symmetric functions. It is well known that ϕ_{MI} and $\phi_{\min}$ are concave functions. Using Lemma 5.2 it then follows that these functions are Schur-concave. Using standard calculus, the partial derivatives of the remaining utility functions (5.7) with respect to γ_m are

$$\begin{aligned}
\phi'_{\mathrm{PT-ZF},m}(\boldsymbol{\gamma}) &= \frac{1}{\gamma_m^2} f_1(\boldsymbol{\gamma}),\\
\phi'_{\mathrm{PT-MMSE},m}(\boldsymbol{\gamma}) &= \frac{1}{(1+\gamma_m)^2} f_2(\boldsymbol{\gamma}),\\
\phi'_{\mathrm{OFDM-BER},m}(\boldsymbol{\gamma}) &= \frac{\exp\left(-\frac{\gamma_m}{4}\right)}{\sqrt{\gamma_m}} f_3(\boldsymbol{\gamma}),\\
\phi'_{\mathrm{OFDM-BLER},m}(\boldsymbol{\gamma}) &= \frac{\exp\left(-\gamma_m/4\right)}{\sqrt{\gamma_m}\left(1-Q\left(\sqrt{\gamma_m/2}\right)\right)} f_4(\boldsymbol{\gamma}).
\end{aligned}$$

Here, we define $f_1 = M\left(\sum_i \gamma_i^{-1}\right)^{-2}$, $f_2 = M^{-1}(1-\delta)^{-2}$, $f_3 = 1/(2\sqrt{\pi})$ and $f_4 = \phi_{\mathrm{OFDM-BLER}}/(2\sqrt{\pi})$. Clearly, f_1, f_2, f_3, f_4 are all symmetric functions of $\boldsymbol{\gamma}$, so it can be easily verified that (5.16b) is valid. Using Lemma 5.1, $\phi_{\mathrm{PT-ZF}}$, $\phi_{\mathrm{PT-MMSE}}$, $\phi_{\mathrm{OFDM-BER}}$ and $\phi_{\mathrm{OFDM-BLER}}$ are thus Schur-concave functions. □

5.5.2 Algorithm 5.2 for Problem Multiple ARQ-SA

We now establish that Problem Multiple ARQ-SA is at least NP-hard for the utility function $\phi_{\min}$. To do so, we relate this problem with a simpler problem that is known to be NP-hard. In [102], the following task allocation problem is considered: assign N_0 tasks with processing time $\boldsymbol{\beta}$ to M_0 processors, so that the minimum processor time required to complete all the tasks is maximized. This is analogous to Problem Multiple ARQ-SA: assign N_0 ARQ subcarriers with SNRs $\boldsymbol{\beta}$ to M_0 original subcarriers, so that the minimum SNR across all effective SNRs is maximized. In the task allocation problem, however, the processor are identical, while in Problem Multiple ARQ-SA, the original subcarrier cannot be treated identically since their SNRs $\boldsymbol{\alpha}$ may not be the same. If $\boldsymbol{\alpha} = \alpha\mathbf{1}$ for some α, our problem is exactly equivalent to the task allocation problem for any $\boldsymbol{\beta}$. This implies that knowing the optimal solution for our problem allows the task allocation problem to be solved, but not vice versa. Our problem is therefore harder. Since the task allocation problem is NP-hard [102], multiple ARQ-SA is at least NP-hard. Algorithm 5.2, which can be implemented with a complexity of $O(M^2)$, is therefore not likely to be optimal for solving Problem Multiple ARQ-SA.

	M	32	64	1024
Algorithm 5.2:	$\mathcal{N}_{\mathrm{m}}/M$	5.00	6.00	10.00
Algorithm 5.1:	$\mathcal{N}_{\mathrm{s}}/M$	3.68	4.62	8.56
Algorithm 5.3, $G = 2$:	$\mathcal{N}_{\mathrm{g}}/M$	1.38	1.84	3.78
Algorithm 5.3, $G = 4$:	$\mathcal{N}_{\mathrm{g}}/M$	0.48	0.69	1.64

Table 5.1: Amount of signalling required in bit per subcarrier for Algorithms 5.1, 5.2, 5.3 and various number of subcarriers M. We fix $M = M_0 = N_0$.

We remark that the LPT (longest processing time first) algorithm considered in [102] is a special case of Algorithm 5.2; both are equivalent if $\boldsymbol{\alpha} = \alpha\mathbf{1}$.

A numerical counter-example confirms that Algorithm 5.2 is sub-optimal for *all* utility functions (5.7). Let $\boldsymbol{\alpha} = [2, 1, 6]^T, \boldsymbol{\beta} = [2, 3, 4]^T$. Using Algorithm 5.2 leads to an ARQ-SA given by (5.12), and so $\boldsymbol{\gamma}^* = [7, 5, 6]^T$. However, if we interchange the first two rows of (5.12), we obtain $\boldsymbol{\gamma}^o = [6, 6, 6]$. Clearly, $\boldsymbol{\gamma}^o \prec \boldsymbol{\gamma}^*$ according to the definition of majorization, and so $\phi(\boldsymbol{\gamma}^o) \geq \phi(\boldsymbol{\gamma}^*)$ for all Schur-concave functions. It can be easily verified that $\phi(\boldsymbol{\gamma}^o) \neq \phi(\boldsymbol{\gamma}^*)$ for all utility functions (5.7). Hence, we have $\phi(\boldsymbol{\gamma}^o) > \phi(\boldsymbol{\gamma}^*)$ and thus we conclude that Algorithm 5.2 is sub-optimal.

5.6 Grouping of Subcarriers

The assignment obtained by Algorithm 5.1 or Algorithm 5.2 can be determined by the transmitter and made known to the receiver, or determined by the receiver and made known to the transmitter. Independent of the mechanism actually used, we propose a method to reduce the amount of signaling information required to convey the assignment.

5.6.1 Amount of Signalling Required

For Algorithm 5.1, each ARQ subcarrier is assigned to a different original subcarrier. Since there are N_0 ARQ subcarriers (excluding virtual subcarriers) and M_0 original subcarriers, there are in total $P_{N_0}^{M_0} = M_0!/(M_0 - N_0)!$ possible permutations. Hence, $\mathcal{N}_{\mathrm{s}} = \log\left(P_{N_0}^{M_0}\right)$ bits of signalling are required to communicate a chosen assignment. We use base 2 for all logarithms. For Algorithm 5.2, each ARQ subcarrier can be assigned to any original subcarrier. There are in total ${M_0}^{N_0}$ possible assignments. Hence, $\mathcal{N}_{\mathrm{m}} = N_0 \log(M_0)$ bits of signalling are required to communicate a chosen assignment.

For many applications, the channel is quasi-static over a period of time. The assignment needs only to be updated every L OFDM symbols (say), when the channel has sufficiently changed. Typically, L is in the order of hundreds in wireless LAN

applications. A useful measure of the overhead used is therefore the *fractional signalling overhead* (FSO) given by $\mathcal{N}/(M_0 L)$ in bit per subcarrier per update, where $\mathcal{N} = \mathcal{N}_\mathrm{s}$ for single ARQ-SA and $\mathcal{N} = \mathcal{N}_\mathrm{m}$ for multiple ARQ-SA. For simplicity, let $M_0 = N_0 = M$. Table 5.1 shows the amount of signalling required for Algorithms 5.1 and 5.2 for $L = 1$.

Obviously, $\mathcal{N}_\mathrm{m} \geq \mathcal{N}_\mathrm{s}$. We consider the saving in FSO for using Algorithm 5.1 compared to Algorithm 5.2 for large M. By using Stirling's formula [103], we have $M! = \sqrt{2\pi M}\,(M/\mathrm{e})^M\,(1 + O(1/M))$ and so $\mathcal{N}_\mathrm{s} = \log(M!) = M\log M - M\log(\mathrm{e}) + O(\log(M))$. Since $\mathcal{N}_\mathrm{m} = M\log M$, we obtain the saving as

$$\frac{\mathcal{N}_\mathrm{m} - \mathcal{N}_\mathrm{s}}{ML} = \frac{\log(\mathrm{e})}{L} + O\left(\frac{\log(M)}{ML}\right). \tag{5.18}$$

which approaches $\log(\mathrm{e})/L \approx 1.44/L$ for large M. This shows that using Algorithm 5.1 can lead to a significant saving. Table 5.1 numerically confirms that as M increases, the difference in FSO approaches 1.44 with $L = 1$. Nevertheless, we see that the absolute amount of signalling is still quite large for Algorithm 5.1 and Algorithm 5.2, hence more efficient algorithms are desirable.

5.6.2 Method of Grouping

To reduce the number of possible subcarrier assignments, we can group G contiguous subcarriers and apply Algorithm 5.1 or Algorithm 5.2 on these groups. For each group, we use a group-equivalent SNR to represent the SNR of the group, e.g., the minimum SNR, arithmetic mean or geometric mean of the group. If G is much smaller than the coherence bandwidth of the channel, the subcarriers within each group have approximately the same SNR. Performing assignment based on these grouped subcarriers would likely result in only negligible performance loss. For each group of G ARQ subcarriers that has been assigned to a group of G original subcarriers, a second (deeper) level of assignment is carried out for every subcarrier. This level of assignment may be performed dynamically during run time depending on the CSI, but at the expense of incurring additional signalling bandwidth.

Although the technique of grouping applies to both multiple ARQ-SA and single ARQ-SA, we focus on the latter since our main motivation is to reduce the amount of signalling. For simplicity, we assume that M_0, N_0 is divisible by G. Simulations reveal that using the *minimum SNR* as the group-equivalent SNR, and a *fixed* (arbitrary) assignment for the second-level of assignment typically gives good BER performance. Hence, we propose Algorithm 5.3, as follows.

Algorithm 5.3 For solving Problem Single ARQ-SA.

First-level of assignment:

- Group the original and ARQ subcarrier into groups of G contiguous subcarriers. Each group uses the minimum SNR in the group as the group-equivalent SNR.
- Apply Algorithm 5.1, but with $\boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\beta}$ replaced by their group-equivalent counterparts.
  That is, $\mathbf{A}$ becomes $\widetilde{M}_0 = M_0/G$ by $\widetilde{N}_0 = N_0/G$ matrix.

Second-level of assignment (independent of channel):

- For each group of G ARQ subcarriers assigned to a group of G original subcarriers, assign the ARQ subcarrier with the gth smallest subcarrier index to the original subcarrier with the gth smallest subcarrier index for $g = 1, \cdots, G$.

Algorithm 5.3 reduces to Algorithm 5.1 when $G = 1$.

We denote the amount of signalling required for Algorithm 5.3 as $\mathcal{N}_{\rm g}$. Since the second level of assignment is fixed and does not require signalling, we get $\mathcal{N}_{\rm g} = \log\left(P_{\widetilde{N}_0}^{\widetilde{M}_0}\right)$, by replacing M_0, N_0 in $\mathcal{N}_{\rm s}$ with $\widetilde{M}_0, \widetilde{N}_0$, respectively. Let $M_0 = N_0 = M$. We consider the saving in FSO with grouping compared to no grouping for large M. By using Stirling's formula again, we obtain the saving as

$$\frac{\mathcal{N}_{\rm s} - \mathcal{N}_{\rm g}}{ML} = \frac{\log(M)\,(1 - 1/G)}{L} + O\left(\frac{\log(M)}{M}\right)$$

which approaches $\log(M)\,(1 - 1/G)\,/L$ for large M. As expected, the saving is zero when no grouping is carried out, i.e., when $G = 1$. For large M, we observe that the saving increases slowly (due to the logarithm) with M, but increases relatively quickly with G even for small G. Table 5.1 shows the values of $\mathcal{N}_{\rm g}$ for $G = 2, 4$, which numerically supports the above conclusions.

For illustration, consider $L = 100, M = 64$, and using Algorithm 5.3 with $G = 2$. From Table 5.1, the FSO is $1.84/L = 1.84\%$. To offer a strong error protection for the signalling bits, one may use additional redundancy bits, but overall the amount of overhead remains small.

5.7 Throughput

This section obtains the throughput of the ARQ system based on the transmission structure shown in Fig. 5.2(a). We recall from Section 5.2.2 that an ARQ round consists of an original transmission and an additional ARQ transmission if the original transmission is erroneous, i.e., if a NACK is received. An example of the transmissions over time is shown in Fig. 5.4, where thicker vertical lines mark the boundaries of the ARQ rounds. Let t_i be the discrete time at the end of ARQ round i, where $i = 1, 2, \cdots$. By definition, $t_0 = 0$. Without loss in generality, we assume that each transmission takes one unit time. For example, $t_1 = 1, t_2 = 3, t_3 = 5, t_4 = 6$ in

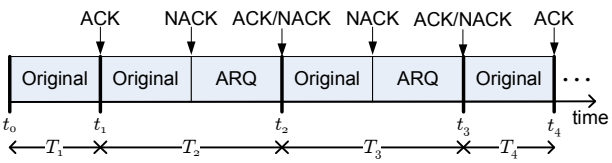


Fig. 5.4: The original transmission and ARQ transmission over time. The random variables t_i and T_i denotes the end and the period of ARQ round i, respectively.

Fig. 5.4. The duration of ARQ round i is given by $T_i = t_i - t_{i-1}$. Since the data symbols transmitted in different ARQ rounds are decoded independently, T_i is i.i.d. and hence a renewal process.

Let s_i be the number of bits recovered in ARQ round i, which takes the role of a *reward* for utilizing T_i units of time. The throughput is given by the total number of recovered bits normalized by the total time spent, over an infinite time horizon. We note from Section 5.2.2 that the transmissions in one ARQ round are independent of other ARQ rounds, and hence the renewal-reward theorem [70, 104] applies. The throughput can hence be determined as

$$S \triangleq \lim_{K\to\infty} \frac{1}{M_0} \frac{\sum_{i=1}^{K} s_i}{\sum_{i=1}^{K} T_i}, \tag{5.19}$$

$$= \frac{1}{M_0} \frac{\mathbb{E}[s]}{\mathbb{E}[T]} \qquad \text{with probability one.} \tag{5.20}$$

The throughput can now be computed if the expectations of T and s are known (the indices are dropped for brevity). To this end, it is convenient to define the following error events. Let

- $\mathsf{E}_1(M_0)$ be the event that at least one original DS is erroneous, after decoding from M_0 original DSs (equivalently the event that the original transmission fails);
- $\mathsf{E}_2(M_0, N_0)$ be the event that at least one original DS is erroneous, after *jointly* decoding from M_0 original DSs and N_0 incremental RSs;
- $\mathsf{E}_3(M_1, N_1)$ be the event that at least one *new* data symbol is erroneous, after *jointly* decoding from M_1 ARQ DSs and N_1 ARQ RSs.

For brevity, we drop these arguments. The complementary event of E_i is denoted as E_i^c. For example, E_1^c is the event that *all* the original DSs are successfully decoded in the original transmission.

Let n_b be the number of bits transmitted in each data symbol. Since QPSK modulation is used, we have $n_\mathrm{b} = 2$. Since we employ error detection on a per block basis, none of the bits in a block (of original DSs or ARQ DSs) is considered to be recovered if at least one of the data symbols is erroneous; otherwise, all the bits in the block are recovered. The relationship of T and s/n_b can then be represented as shown in Fig. 5.5. The ARQ round begins with an original transmission. This original transmission is successful (i.e., event E_1^c occurs) with probability $1 - \Pr(\mathsf{E}_1)$, for which the

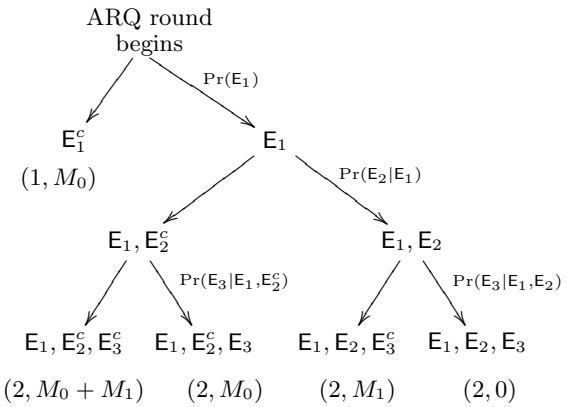


Fig. 5.5: Probability tree over the original and ARQ transmissions in an ARQ round. Below all possible events that result in the termination of the ARQ round, we provide $(T, s/n_\mathrm{b})$ indicating the duration and normalized throughput of that ARQ round.

ARQ round then terminates with $T = 1, s/n_\mathrm{b} = M_0$. On the other hand, the original transmission is erroneous with probability $\Pr(\mathsf{E}_1)$, and an ARQ transmission will be sent, hence $T = 2$. Depending on whether E_2 occurs and whether E_3 occurs, s/n_b can then take four possible values: $M_0 + M_1, M_0, M_1$ or 0, as shown in Fig. 5.5. From Fig. 5.5, we obtain

$$\mathbb{E}[T] = \Pr(\mathsf{E}_1^c) + 2\Pr(\mathsf{E}_1) = 1 + \Pr(\mathsf{E}_1), \tag{5.21}$$

$$\begin{aligned}\mathbb{E}[s]/n_\mathrm{b} &= M_0\left(\Pr(\mathsf{E}_1^c) + \Pr(\mathsf{E}_1, \mathsf{E}_2^c)\right) + M_1\left(\Pr(\mathsf{E}_1, \mathsf{E}_2^c, \mathsf{E}_3^c) + \Pr(\mathsf{E}_1, \mathsf{E}_2, \mathsf{E}_3^c)\right)\\ &= M_0\left(1 - \Pr(\mathsf{E}_1, \mathsf{E}_2)\right) + M_1 \Pr(\mathsf{E}_1)\left(1 - \Pr(\mathsf{E}_3|\mathsf{E}_1)\right).\end{aligned} \tag{5.22}$$

So, the throughput (5.20) becomes

$$S(\bar{\gamma}, M_0, N_0, M_1, N_1) = n_\mathrm{b}\frac{1 - \Pr(\mathsf{E}_1, \mathsf{E}_2) + \frac{M_1}{M_0}\Pr(\mathsf{E}_1)\left(1 - \Pr(\mathsf{E}_3|\mathsf{E}_1)\right)}{1 + \Pr(\mathsf{E}_1)}. \tag{5.23}$$

The throughput depends, via the error events, on the average SNR $\bar{\gamma} \triangleq \mathbb{E}[\alpha] = \mathbb{E}[\beta]$, and on the size of the subcarrier sets M_0, N_0, M_1, N_1. This dependence is written explicitly in (5.23).

We now consider the scenario that full redundancy is used, i.e., $M_0 = N_0$ and $M_1 = N_1 = 0$. This is a typical case considered in the literature, e.g., [39–41, 70]. The throughput is then upper bounded as

$$S(\bar{\gamma}, M_0, M_0, 0, 0) \le S^*(\bar{\gamma}; M_0) \triangleq \frac{n_\mathrm{b}}{1 + \Pr(\mathsf{E}_1)} \tag{5.24}$$

since $\Pr(\mathsf{E}_1, \mathsf{E}_2) \ge 0$. If $\Pr(\mathsf{E}_2|\mathsf{E}_1)$ approaches zero, the throughput $S(\bar{\gamma}, M_0, M_0, 0, 0)$ approaches the upper bound $S^*(\bar{\gamma}; M_0)$. This scenario occurs if the effective SNR is high after applying Algorithm 5.1, 5.2 or 5.3, which is typical when the average SNR is high. On the other hand, if we employ incremental redundancy so that some ARQ

DSs are sent, i.e., $M_1 > 0$, it is possible for the throughput to exceed $S^*(\bar{\gamma}; M_0)$. Specifically, if

$$M_1(1 - \Pr(\mathsf{E}_3|\mathsf{E}_1)) > M_0 \Pr(\mathsf{E}_2|\mathsf{E}_1) \tag{5.25}$$

is satisfied, it can be easily shown that the throughput (5.23) must exceed $S^*(\bar{\gamma}; M_0)$, which is already the largest possible throughput with full redundancy. This implies that not sending any data in the ARQ transmission can limit throughput. In the simulations in Section 5.8, we demonstrate that at high SNR, full redundancy indeed poses as a severe limitation to the significant throughput gain that can otherwise be achieved with incremental redundancy.

Intuitively, the condition (5.25) can be explained as follows. Given that the original transmission is erroneous, i.e., E_1 occurs, the L.H.S of (5.25) reflects the throughput gained by the ARQ DSs after the ARQ transmission, while the R.H.S reflects the throughput lost by the original DSs. If the gain is larger than the loss, then one should transmit ARQ DSs to achieve an overall gain in throughput.

5.8 Numerical Results

We first use a case study to illustrate the difference of our proposed algorithms with optimal solutions. Then, we show the improvement in block error rate (BLER) and throughput achieved.

5.8.1 Case Study

To better illustrate the case study, we evaluate the solutions numerically for a small number of subcarriers with $M_0 = N_0 = 10$, and we use the minimum-SNR utility function $\phi_{\min}$ in (5.7a). The (arbitrary) SNRs of the original and ARQ subcarriers $\boldsymbol{\alpha}, \boldsymbol{\beta}$ that we consider are generated independently from an i.i.d. exponential distribution and are shown in Fig. 5.6. For clarity, the original subcarriers are sorted increasingly according to their SNRs, while the ARQ subcarriers are sorted decreasingly.

To solve Problem Single ARQ-SA and Problem Multiple ARQ-SA optimally, we formulate them as mixed-integer linear programs (MILPs) and solve these MILPs using GLPK [105], a standard linear programming software; we leave out the details.

Algorithms 5.1, 5.2 are also implemented. Fig. 5.6 shows the resulting minimum effective SNRs. All the above four approaches clearly give a significantly higher minimum effective SNR compared to the minimum SNR of the original subcarriers (without ARQ). Moreover, Algorithm 5.2 performs (slightly) better than Algorithm 5.1, which is expected as fewer constraints have been imposed. For single ARQ-SA, the MILP solution gives the same minimum effective SNR as Algorithm 5.1[3], as expected due

[3]The other effective SNRs can be different, because the utility function $\phi_{\min}$ only considers the minimum SNR.

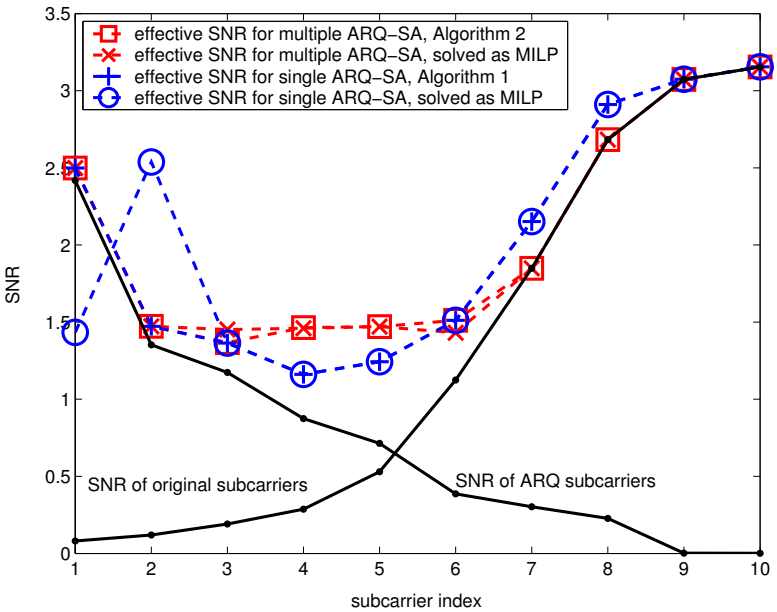


Fig. 5.6: The effective SNRs (not in dB, shown as dotted lines) after applying Algorithm 5.1 and Algorithm 5.2, based on typical SNR realizations of original subcarriers and ARQ subcarriers (shown as full lines). For clarity, the SNRs of the original subcarriers are ordered increasingly, while the ARQ subcarriers are ordered decreasingly.

to Theorem 5.1. For multiple ARQ-SA, the MILP solution is marginally better than Algorithm 5.2. Specifically, the smallest effective SNR based on MILP (at subcarrier 6) is slightly higher than based on Algorithm 5.2 (at subcarrier 3).

We note that using Algorithm 5.1 already improves the worst-case SNR gain significantly, while the additional improvement offered by other solutions is relatively small. Although we study a particular case here, this conclusion holds typically. This is because for wireless channels that are frequency selective, the weakest subcarriers are already significantly boosted by the strongest subcarriers using Algorithm 5.1. Hence, any further gain by using a more sophisticated algorithm is likely to be small.

5.8.2 Performance Evaluation

5.8.2.1 Scenario

For our performance evaluation, we consider a QPSK modulated system and a PT-OFDM system with $M_0 = 64$ subcarriers at different average SNR $\bar{\gamma}$. We use the transform [35]

$$\mathbf{W} = \mathbf{F} \times \mathrm{diag}\{1, \lambda, \cdots, \lambda^{M_0-1}\} \tag{5.26}$$

before applying IFFT, where $\lambda = \exp(-\mathrm{j}\pi/(2M_0))$ and $\mathbf{F}$ is the FFT matrix. Notice that the FFT and the IFFT cancel out, so the transformation is equivalent to rotating the data symbol in the time domain. The transform $\mathbf{W}$ is unitary and has constant-amplitude elements. Although it has been designed for maximum likelihood decoding (MLD) in [35], simulation studies (not shown here) indicate that it also results in good error performance generally when used with other detection schemes. For illustration, we use the ZF equalizer given by (5.6a); more advanced detectors such as MLD [35] or iterative subcarrier reconstruction [100] can also be used. We model the wireless channel with $L_h + 1 = 16$ i.i.d. time domain channel taps. The channel used for transmission is assumed to be time invariant in each ARQ round. We assume an error-free channel is available to signal the ACK bit and the assignment used.

The ARQ subcarriers are classified as incremental RSs, ARQ DSs or ARQ RSs via subcarrier sets $\mathcal{S}_1, \mathcal{S}_0', \mathcal{S}_1'$, respectively; see Fig. 5.2(a). In our simulations, we select $\mathcal{S}_1$ as N_0 subcarriers that are roughly spaced uniformly apart, so that these subcarriers experience independent channels as much as possible. From the remaining subcarriers, similarly we select $\mathcal{S}_0'$ as M_1 subcarriers that are spaced uniformly apart. Finally, $\mathcal{S}_1'$ is made up of the remaining subcarriers. Unlike [99], we do not reserve stronger ARQ subcarriers only for $\mathcal{S}_1$, since this will result in weaker subcarriers for $\mathcal{S}_0'$ and $\mathcal{S}_1'$.

5.8.2.2 Block Error Rate (BLER)

We consider these BLERs that are used to calculate the throughput S in (5.23):

- $\Pr(\mathsf{E}_1)$, the BLER for the original DSs without using ARQ,

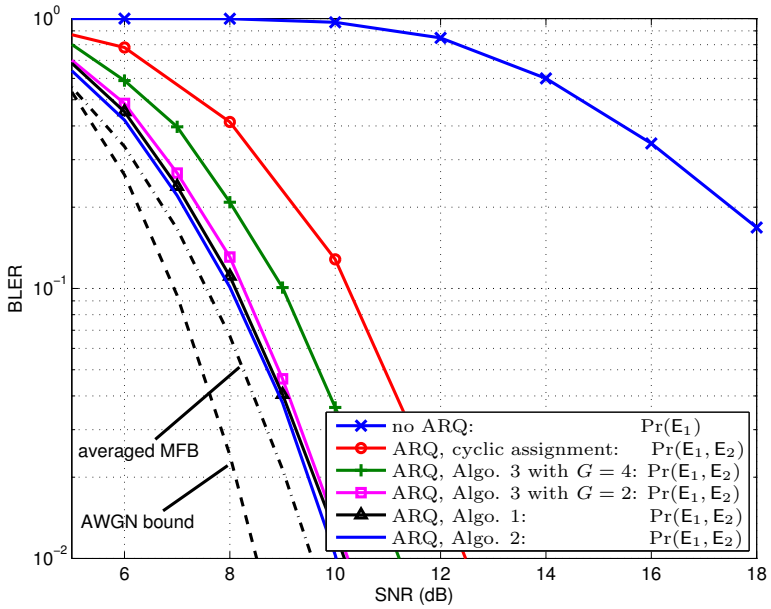


Fig. 5.7: BLER performance using full redundancy, $M_0 = N_0 = 64$.

- $\Pr(\mathsf{E}_1, \mathsf{E}_2)$, the BLER for the original DSs by using ARQ, and
- $\Pr(\mathsf{E}_3|\mathsf{E}_1)$, the BLER for the ARQ DSs, given that the original transmission fails.

Monte Carlo simulations are used to obtain the BLERs for Algorithms 5.1, 5.2 and 5.3, as analytical expressions for the BLERs involve order statistics and often do not yield closed-form results.

For benchmarking, we use a (fixed) cyclic assignment considered in the literature [39–41], in which CSI is not exploited. In this cyclic assignment, we cyclically shift the indices of the ARQ subcarrier with respect to the original subcarriers by 16 subcarriers, which allows each DS and its corresponding RS to experience channels that are close to independent.

Lower bounds based on idealistic conditions are used to check how good our schemes perform. Since much of the throughput gain comes from the original DSs, lower bounds are considered only for $\Pr(\mathsf{E}_1, \mathsf{E}_2)$. First, we use the *averaged matched filter bound* (MFB) to provide a lower bound, based on the same frequency-selective channel model. We assume that there is no interference from other data symbols and hence a matched filter is used for detection; moreover, ARQ is always activated. Second, we employ the *AWGN bound*, where an AWGN channel is used, i.e., the original and ARQ subcarriers do not experience fading. Moreover, ARQ is always activated. Details of both bounds are provided in Appendix 5.B.

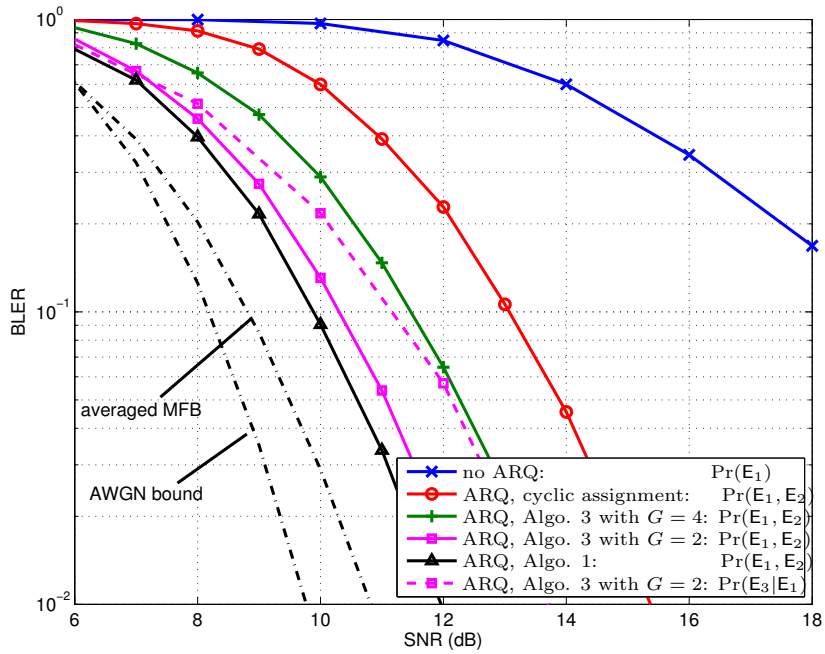


Fig. 5.8: BLER performance of the original DSs using incremental redundancy, with $M_0 = 64, N_0 = 32$. An illustrative case of the BLER $\Pr(\mathsf{E}_3|\mathsf{E}_1)$ when ARQ DSs are sent, where $M_1 = 24, N_1 = 8$, is also shown.

$N_0 = 64$: We first consider that full redundancy is employed, i.e., $M_0 = N_0 = 64$ and $M_1 = N_1 = 0$; see Fig. 5.2(b). Since ARQ DS is not sent, $\Pr(\mathsf{E}_3|\mathsf{E}_1)$ is not relevant here. From Fig. 5.7, the BLER $\Pr(\mathsf{E}_1, \mathsf{E}_2)$ obtained from using Algorithm 5.1 is almost the same as Algorithm 5.2. Both algorithms perform substantially better than the BLER $\Pr(\mathsf{E}_1)$ when ARQ is not used, and provide more than 2 dB of SNR gain compared to using a cyclic assignment. Surprisingly, Algorithm 5.1 and Algorithm 5.2 are only about 0.5 dB away from the averaged MFB bound, even though only a simple ZF equalizer has been employed.

Finally, we observe that using Algorithm 5.3 with grouping of $G = 2$ achieves a performance that is only a few tenths of dBs away from that of Algorithm 5.1 (without grouping). When $G = 4$, the BLER is about 1 dB away from Algorithm 5.1, but is still about 1 dB better than using cyclic permutation. This suggests that using grouping of $G = 2$ is adequate for this scenario; if further reduction of signalling is desired, using grouping of $G = 4$ can be a good compromise.

$N_0 = 32$: Next, we consider $N_0 = 32$ in Fig. 5.8. Since $N_0 < M_0$, incremental redundancy is strictly used (instead of full redundancy); see Fig. 5.2(a). We first consider the BLER for the original DSs $\Pr(\mathsf{E}_1, \mathsf{E}_2)$. The performance of Algorithm 5.2 is very

close to that of Algorithm 5.1 and is not shown. As in Fig. 5.7, Fig. 5.8 shows that using more signalling for the assignment leads to better performance. However, these performance gaps between different algorithms widen for $N_0 = 32$ in Fig. 5.8, compared to $N_0 = 64$ in Fig. 5.7. In particular, the gaps between cyclic permutation and proposed algorithms are more significant. Hence, when redundancy is limited, the proposed algorithms become more important in maintaining a reasonable system performance. We now consider the BLER for the ARQ DSs $\Pr(\mathsf{E}_3|\mathsf{E}_1)$, by using Algorithm 5.3 with $G = 2$. We transmit $M_1 = 24$ ARQ DSs and $N_1 = 8$ ARQ RSs; these parameters have been optimized to maximize the throughput, as explained in Section 5.8.2.3. At a sufficiently high SNR of 12 dB, for example, the original DSs are received with low error probability, yet we can additionally send ARQ DSs with an error probability of only around 0.06.

5.8.2.3 Throughput

Fig. 5.9 illustrates the throughput obtained with and without ARQ. We focus on using Algorithm 5.3 with $G = 2$, which gives good performance with small overhead. With full redundancy, we observe that a substantial improvement is obtained compared to when ARQ is not used. However, we note that this improvement is limited by the upper bound S^* given in (5.24) for SNR$\leq$ 10 dB. To obtain further improvement at high SNR, incremental redundancy must therefore be used. This implies that the number of incremental RSs N_0 should be reduced, but this has little effect at high SNR since the original DSs can usually still be recovered.

So far we have fixed N_0, M_1, N_1. We now optimize these parameters with the use of incremental redundancy to maximize the throughput for SNR $\geq$ 10 dB. In our simulations, we vary the parameters in steps of four and obtain the maximized throughput as shown in Fig. 5.9, where the optimized parameters are indicated as $[N_0, M_1, N_1]$. We observe that a significant gain, up to about 4 dB, can be realized at high SNR. On the other hand, in the low-SNR regime, full redundancy should preferably be used to ensure reliable packet recovery. To improve throughput in this regime, an option is to increase the number of ARQ transmissions to more than one, but at the expense of incurring higher delay.

5.9 Conclusion

We propose two ARQ schemes based on subcarrier assignment for general OFDM systems with a possible pre-transform. For the single ARQ-SA scheme, we propose an optimum subcarrier assignment algorithm that optimizes the class of Schur-concave utility functions. For the multiple ARQ-SA scheme, we propose a sub-optimum algorithm. In order to keep the amount of feedback required to communicate the assignment low, we consider subcarrier grouping techniques by grouping contiguous subcarriers and performing subcarrier assignment on these groups. Numerical results have shown an improvement of the error performance even when limited redundancy

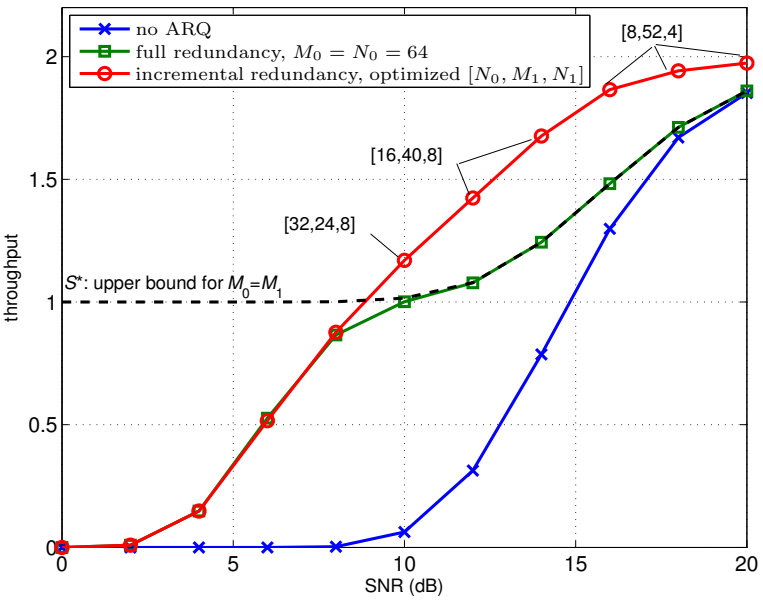


Fig. 5.9: Throughput using Algorithm 5.3 with subcarrier grouping of $G = 2$.

is available for ARQ, which has led to significant throughput gains over a wide range of SNR. Even though in this chapter we restrict to the case of at most one retransmission, our ARQ schemes can be generalized easily to any number of retransmissions.

Appendix 5.A ARQ-SA for Two and More ARQ Transmissions

We show that the subcarrier assignment problem for the original DSs for two ARQ transmissions is fundamentally the same as for one ARQ transmission, and so both problems can be solved similarly.

Consider that a second ARQ transmission is activated. In this second ARQ transmission, suppose that the subcarrier set $\mathcal{S}_2$ is available to provide redundancy for the original DSs. Let $\{\beta'_n, n \in \mathcal{S}_2\}$ be the SNRs corresponding to $\mathcal{S}_2$, and let $\mathcal{A}'(m) \in \mathcal{S}_2$ be the assignment of the ARQ subcarriers for the mth original subcarrier. After MRC, the effective SNR at subcarrier m becomes

$$\gamma_m = \alpha_m + \sum_{n \in \mathcal{A}(m)} \beta_n + \sum_{n \in \mathcal{A}'(m)} \beta'_n,\ m \in \mathcal{S}_0. \tag{5.27}$$

Compared to (5.5), we have included the additional contribution of the second ARQ transmission. Due to causality, past assignments $\{\mathcal{A}(m)\}$ are fixed before the current assignments $\{\mathcal{A}'(m)\}$ are carried out. Consequently, $(\alpha_m + \sum \beta_n)$ can be treated as a single *fixed* term, similar to α_m which is fixed in (5.5). Hence, the subcarrier assignment problem, either of selecting $\{\mathcal{A}'(m)\}$ given (5.27), or of selecting $\{\mathcal{A}(m)\}$ given (5.5), is fundamentally the same problem and can be solved similarly. This conclusion also holds for more than two ARQ transmissions by treating all previous assignments as fixed (and similarly if we consider the assignment for the ARQ DSs instead of the original DSs).

Appendix 5.B Bounds for ARQ in Fading Channels

Typically, the MFB gives a lower bound for an error probability *without ARQ for a AWGN channel*. Here, we modified it for ARQ systems in a fading channel. To this end, we assume that ARQ is always activated and we take the expectation of the error probability over the fading channel.

Consider a unitary transform with constant-amplitude elements, such as (5.26), where $|w_{mn}|^2 = 1/M_0$. To obtain a lower bound, without loss of generality we assume that only the first symbol is transmitted, i.e., $\mathbf{x} = [x_1, \mathbf{0}_{M_0-1}]$. This symbol is transmitted over the original subcarriers *and* the ARQ subcarriers with SNRs $\boldsymbol{\alpha}, \boldsymbol{\beta}$, respectively. A matched filter is used at the receiver, which equivalently collects the SNR over all original and ARQ subcarriers. Thus, the equivalent SNR for x_1 is $\gamma_{\mathrm{MFB}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \left(\sum_{m \in \mathcal{S}_0} \alpha_m + \sum_{n \in \mathcal{S}_1} \beta_n\right)/M_0$. For QPSK modulation, the BER is $P_{\mathrm{e}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = Q\left(\sqrt{\gamma_{\mathrm{MFB}}(\boldsymbol{\alpha}, \boldsymbol{\beta})/2}\right)$. Since any bit error constitutes a block error and there are $2M$ bits in a block, the BLER is

$$\mathrm{BLER}_{\mathrm{MFB}} = \mathbb{E}_{\boldsymbol{\alpha},\boldsymbol{\beta}}\left[1 - (1 - P_{\mathrm{e}}(\boldsymbol{\alpha}, \boldsymbol{\beta}))^{2M}\right]. \tag{5.28}$$

Here, the expectation is performed over $\boldsymbol{\alpha}, \boldsymbol{\beta}$. A semi-analytical method can be used to obtain numerical results, by averaging the term within the expectation operator over realizations of $\boldsymbol{\alpha}, \boldsymbol{\beta}$ generated by Monte Carlo simulations.

The AWGN bound provides a looser bound, but in closed form. It is obtained similarly as the MFB, except that the channel is always fixed as the average SNR, i.e., $\alpha_m = \beta_n = \bar{\gamma}$ for all m, n. Thus, the equivalent SNR for x_1 becomes $\gamma_{\mathrm{MFB}} = \bar{\gamma}(M_0 + N_0)/M_0$, a constant. Hence, the BLER provided by the AWGN bound is given by $\mathrm{BLER}_{\mathrm{AWGN}} = 1 - \left(1 - Q\left(\sqrt{\gamma_{\mathrm{MFB}}/2}\right)\right)^{2M}$.

CHAPTER 6

SUCCESSIVE-CAPTURE ANALYSIS OF RTS/CTS

This chapter[1] considers a wireless ad-hoc network with random ALOHA transmissions between nodes. However nodes can decide to use the Request-to-Send (RTS) / Clear-to-Send (CTS) protocol for occasional long packets. Via the RTS/CTS protocol, nodes proactively make a temporal and spatial reservation of the channel before sending the actual data payload (DATA). The aim of this chapter is to explore the effectiveness of the RTS/CTS protocol in such a wireless network, in particular by relaxing the commonly made assumption that an entire RTS/CTS/DATA cycle is always successful if only the RTS packet is recovered. Our results show that the evaluation of the link throughput becomes inaccurate under this assumption if the data rate is optimized for the link throughput or if one aims for allowing dense frequency reuse. We quantify the spatial and temporal impact of the RTS/CTS reservation on the network traffic, as well as the link throughput achieved. Numerical results demonstrate that for sufficiently long packets, a significant throughput gain can be achieved by employing the RTS/CTS reservation.

[1]A large part of this work has been published as "Successive-capture analysis of RTS/CTS in ad-hoc networks" in *IEEE Trans. Wireless Commun.*, vol. 7, no. 1, pp. 213–223, Jan. 2008.

6.1 Introduction

Ad-hoc networks can provide wireless connectivity even in the absence of a fixed network infrastructure [106]. In such networks, distributed peer-to-peer communications are carried out typically over a common frequency band. Fig. 6.1(a) on the next page shows an ad-hoc network with source-destination pairs $\mathsf{S} \rightarrow \mathsf{D}$ and $\mathsf{S}_i \rightarrow \mathsf{D}_i$. Each arrow depicts the direction of transmission of data payload (double arrow) or signaling overhead (single arrow) from a sender to a receiver.

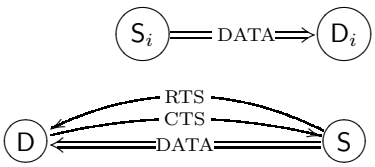
In an infrastructure-based wireless network, by comparison, an access point (AP) acts as the gateway to the infrastructure. Mutual interference is reduced by appropriately assigning different frequencies to neighboring APs. This assignment, however, is usually performed in an ad-hoc fashion in the home wireless LANs, as compared to the centralized planning used commonly in traditional cellular networks. Wireless nodes, known as stations (STAs), send or receive data payload (DATA) packets via the AP in a cell. In a downlink, the STA is the destination, while the AP is the source. In an uplink, their roles are reversed, see Fig. 6.1(b). Here, two STAs depicted as sources S_i and S compete to send their DATA to the AP, their common destination D.

6.1.1 MAC Protocols

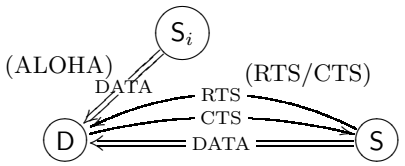
6.1.1.1 Infrastructure Networks

To support a variable number of active nodes, a random access MAC protocol is employed. Well-established random access protocols that are used in infrastructure networks include ALOHA [43], physical carrier sensing [44] and virtual carrier sensing – also known as the Request-to-Send/Clear-to-Send (RTS/CTS) protocol [46, 47]. In the ALOHA protocol, a node transmits whenever new data arrives, or when this data needs to be retransmitted because of a missing acknowledgement (ACK). In the slotted ALOHA variant, transmissions are also globally synchronized. Although simple to implement, ALOHA suffers from excessive interference due to conflicting transmissions when the network load is high [45]. Collisions can be mitigated by physical carrier sensing, where a node is inhibited to transmit if, prior to transmission, it detects signal power from ongoing transmissions. However, physical carrier sensing suffers from the well-known *hidden-node* problem. The RTS/CTS protocol can partly solve this problem.

The RTS/CTS protocol was first proposed as the MACA protocol [46] where physical carrier sensing is replaced by virtual carrier sensing: the source first informs the destination of its intention to exchange data by issuing an RTS packet, and the destination confirms this with a CTS packet, after which the source sends the DATA packet. All other nodes (including hidden nodes) that recover the RTS or CTS packet are inhibited from transmitting during some specified time interval, to facilitate a successful RTS-CTS-DATA cycle. This *RTS/CTS cycle* can be extended by an ACK



(a) Ad-hoc network with peer-to-peer transmissions, where sources S, S_i send DATA to D, D_i, respectively.



(b) Infrastructure network with uplink transmissions, where sources S, S_i, known as stations, send DATA to a common destination D, known as an access point.

Fig. 6.1: ALOHA and RTS/CTS protocols in different types of networks. Each arrow depicts the direction of transmission of data payload (double arrow) or signaling overhead (single arrow) from a transmitter to a receiver.

packet from the destination to reduce the delay caused by erroneous cycles at the transport layer [47]. In Fig. 6.1(b), the ALOHA protocol is used by S_i to send data to D. To protect longer packets, the RTS/CTS protocol is used by S to send data to D.

Current IEEE 802.11 systems are mostly set up as infrastructure networks, where the RTS/CTS protocol is implemented in the so-called distributed co-ordination function (DCF) [48] . An idealized timing model of the protocol is given in Fig. 6.2. An ACK phase can be additionally appended to confirm that data has been recovered.

In an infrastructure network, the STAs only communicate with the AP, so only a single transmission can be successful at any time. Spatial reuse is not required within each cell. Consequently, the lowest possible rates are used to encode the RTS and CTS packets so that the inhibition practically covers the entire cell area. Further, physical carrier sensing is always used concurrently with RTS/CTS in IEEE 802.11.

6.1.1.2 Ad-hoc Networks

In an ad-hoc network, multiple simultaneous but spatially separated transmissions are possible, so an extensive inhibition increases the number of exposed nodes, limits spatial reuse and can lead to dead locks [49]. As such, the IEEE 802.11 DCF is not suitable for an ad-hoc network, see for example [49, 50]. Preferably, other transmissions should be allowed as long as they do not interfere harmfully with the RTS/CTS cycle. Compared to an infrastructure network with interference coming only within

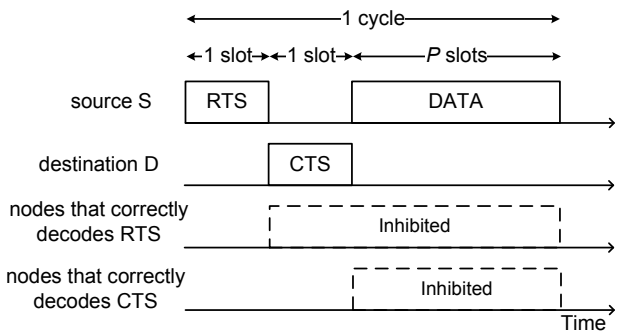


Fig. 6.2: An idealized RTS/CTS protocol. Here, an ACK can be additionally sent to inform the destination that DATA is recovered.

a cell, there is more emphasis on a well-designed MAC protocol in an ad-hoc network, and its spatial demarcation of reservation areas becomes an interesting object of study.

For analysis, it is common to assume that the locations of interferers follow a homogeneous Poisson process [45, 52, 53, 107–109]. This assumption can be justified for ad-hoc communications, particularly if nodes belong to separate groups, or in spatially overlapping infrastructure-based networks operated by different owners.

6.1.2 Successive-Capture Analysis

Analysis of a simplified or approximate RTS/CTS model provides insights that complement simulation results, such as in [49, 50]. In the pioneering analysis in [51], a packet was assumed to be recovered if and only if no other concurrent transmission occurs. This packet detection model can be improved by additionally taking into account path losses and fading of all links, including interference propagation paths. Since an exact accounting of all these propagation effects would not be analytically tractable nor yield useful insights, different simplifying models have been proposed in the literature.

In one packet detection model, perfect reception of a packet transmitted from a certain distance is assumed if no other interfering packet transmission has occurred within some fraction of that distance [107]. An improved model is based on the *capture effect* [45, 52, 53, 108, 109]. We say that a packet *captures* a receiver if the signal-to-noise plus interference ratio (SINR) experienced at the receiver is larger than a certain capture ratio. In this model, packets are considered to be recovered (even in the presence of interfering signals) whenever a capture occurs. The RTS/CTS protocol has also been analyzed considering the capture effect under Rayleigh fading in [54], and under both Rayleigh and shadow fading in [55].

So far in the literature [51, 54, 55], the RTS/CTS protocol has been modeled by assuming that if the RTS packet captures the destination, then the CTS, DATA and

ACK packets are always recovered too. This becomes optimistic for environments with dense (or even contiguous) spatial reuse. The boundaries of the inhibited area are fuzzy and harmful transmission from these fringes are likely. Our model considers more realistic, statistically dependent capture probabilities as the protocol progresses; we call this the successive-capture analysis. In particular, it is possible for packets in later phases of the protocol to fail even if the earlier phases have succeeded.

6.1.3 Scenario

In a wireless ad-hoc network, there are many applications in which small amounts of sensor data are sent frequently, while a larger amount of consolidated data is sent occasionally, usually after data aggregation and fusion. An example is in a hospital where routine vital signs of patients are uploaded to a database, while the full medical record of a patient is downloaded by a doctor occasionally, where the data transfer is performed via wireless links. The short, routine messages can be sent efficiently using the ALOHA protocol, while the large amount of record data that requires higher protection can be protected by the RTS/CTS protocol.

In this chapter, we consider an ad-hoc network where (long) packets can be protected by RTS/CTS if the sources deem this advantageous, while for the majority of (short) messages no reservation is made. All nodes listen and obey the inhibitions imposed by the RTS/CTS protocol.

6.1.4 Contributions

The objective of this chapter is to understand, analyze and optimize the RTS/CTS protocol. The effects of the wireless channels are taken into account by employing the successive-capture analysis. We derive a probabilistic description of the time-varying inhibition area from which a reduced amount of interfering traffic arrives as the RTS/CTS protocol progresses. These descriptions lead to capture probabilities in successive phases of the protocol, and eventually to the achieved link throughput. We confirm (and quantify) that for moderately large messages, nodes can achieve a significantly higher throughput when using the RTS/CTS protocol, compared to the ALOHA protocol. From our investigations we further conclude that for given RTS and CTS rates, there is an optimum rate for the DATA packets. Such data rate optimization cannot be obtained without the use of successive-capture analysis, since conventional capture analysis does not adequately account for the effects of data rate on the performance of the RTS/CTS protocol.

The chapter is organized as follows. Section 6.2 specifies the system model. Section 6.3 derives the throughput achieved using the RTS/CTS protocol. Then, Section 6.4 derives the conditional capture probability in each phase of the RTS/CTS cycle. Numerical results are given in Section 6.5. Finally, Section 6.6 concludes the chapter.

6.2 Model

We consider a two-dimensional wireless ad-hoc network operating in a region $\mathcal{A}$, see Fig. 6.1(a). We classify nodes according to their roles as sources or destinations, but in a RTS/CTS cycle the participating source and destination alternatingly act as a *transmitter* or a *receiver*. The $x, y-$coordinates of the tagged source S and tagged destination D are denoted by position vectors $\mathbf{a}_{\mathsf{S}}$ and $\mathbf{a}_{\mathsf{D}}$, respectively. Other transmitting sources $\mathsf{S}_i, i = 1, \cdots, N$, are called *interferers*.

The network is dominated by short transmissions without RTS/CTS, while occasional long transmissions are protected by RTS/CTS. So we may reasonably assume that the RTS/CTS cycles do not overlap with each other, but nodes see interference from randomly arriving messages. Thus we focus on a single RTS/CTS cycle between S to D.

6.2.1 RTS/CTS Protocol

The source S uses the RTS/CTS protocol to send DATA to the destination D, as summarized in Fig. 6.2. Each signalling packet, specifically either the RTS packet or CTS packet, uses one slot while the DATA packet uses P slots. All (potential) interferers obey the inhibitions provided that they recover the RTS, CTS messages. Notations associated with the respective RTS, CTS and DATA phases are denoted with subscripts $\mathrm{R}, \mathrm{C}, \mathrm{D}$. For simplicity, time is slotted and synchronized.

The ACK is omitted to clarify our analytical approach and this allows direct benchmarking with the classical slotted ALOHA [43]. To isolate the effects of virtual carrier sensing, physical carrier sensing is disabled. This approach, also used in the MACA protocol [46], allows a higher spatial reuse to be realized in an ad-hoc network. Our analysis can nevertheless be extended when ACK is included or when physical carrier sensing is used.

If a packet is successfully recovered by the destination, we say that the packet captures the destination. Let $\mathcal{C}_{\mathrm{R}}$ denote the event that a RTS slot captures D, $\mathcal{C}_{\mathrm{C}}$ the event that a CTS slot captures S and $\mathcal{C}_{\mathrm{D}}$ the event that a DATA slot captures D. We note that $\mathcal{C}_{\mathrm{R}}$ ($\mathcal{C}_{\mathrm{C}}$) implies that the RTS (CTS, respectively) packet is also recovered, while $\mathcal{C}_{\mathrm{D}}$ may not imply that the DATA packet, which may consist of multiple slots, is entirely recovered. Since capture events have a sequential and causal relation,

$$\mathcal{C}_{\mathrm{C}} \quad \Leftrightarrow \quad \mathcal{C}_{\mathrm{R}} \cap \mathcal{C}_{\mathrm{C}}; \tag{6.1}$$

$$\mathcal{C}_{\mathrm{D}} \quad \Leftrightarrow \quad \mathcal{C}_{\mathrm{R}} \cap \mathcal{C}_{\mathrm{C}} \cap \mathcal{C}_{\mathrm{D}}. \tag{6.2}$$

We only spell out the full sequences of events in expressions for conditional probabilities when this contributes to the intuition; otherwise we only denote the last event in time. The intersection notation $\cap$ is replaced with a comma for brevity when it is obvious.

6.2.2 Wireless Network Model

6.2.2.1 Traffic

The *traffic* in a slot is fully described by the number and positions of active interferers within an operation region $\mathcal{A}$:

$$\mathbb{T} = \left\{ \mathcal{N}, \left\{ \mathbf{a}_i \in \mathcal{A} \right\}_{i=1}^{\mathcal{N}} \right\}.$$

The density of the traffic, written as $f_{\mathbb{T}}\left(\mathcal{N}, \{\mathbf{a}_i\}_{i=1}^{\mathcal{N}}\right) = \Pr(\mathcal{N}) \Pi_i f_{\mathbf{a}_i}(\mathbf{a}_i | \mathcal{N})$, consists of products of a probability mass function (pmf) and of probability density functions (pdfs). We consider the limiting case where $\mathcal{A}$ grows infinitely large and $\mathcal{N} \to \infty$. Therefore, nonzero throughput is possible only because of the capture effect and a sufficiently steep path loss law.

In the RTS slot, the traffic is denoted as $\mathbb{T}_{\mathrm{R}}$. In the CTS slot, the traffic is denoted as $\mathbb{T}_{\mathrm{C}}$. The DATA phase spans P slots and the traffic for slot k is denoted as $\mathbb{T}_{\mathrm{D}}(k), k = 1, \cdots, P$. Finally, the *cycle traffic*, $\mathbb{T}_{\mathrm{cycle}} \triangleq \{\mathbb{T}_{\mathrm{R}}, \mathbb{T}_{\mathrm{C}}, \mathbb{T}_{\mathrm{D}}(1), \cdots, \mathbb{T}_{\mathrm{D}}(P)\}$, describes the traffic of all the slots in an RTS/CTS cycle. Due to causality (6.1), the CTS traffic $\mathbb{T}_{\mathrm{C}}$ is defined only if the RTS is recovered, and the DATA traffic $\mathbb{T}_{\mathrm{D}}$ is defined only if the RTS and CTS are recovered.

We assume that DATA transmissions are independent, i.e., we ignore possible correlation due to retransmissions [45] or deferred transmissions. This allows us to study the traffic resulting directly from the RTS/CTS inhibitions. For the RTS slot (and also for the ALOHA protocol), for reasons argued before, we describe the traffic using the Poisson process with *traffic intensity* $\mathcal{G}(\mathbf{a})$ packets per time slot per unit area (ppsa), where $\mathbf{a} \in \mathcal{A}$. Specifically, we assume a homogeneous Poisson process with $\mathcal{G}(\mathbf{a}) = G_o$. For the other slots later in the RTS/CTS cycle, we shall argue in Section 6.4 that the traffic is well described by non-homogeneous Poisson processes.

Generally, for a Poisson process, the density of traffic $\mathbb{T}$ is determined by the traffic intensity $\mathcal{G}(\mathbf{a})$, as follows. Within $\mathcal{A}$, $\mathcal{N}$ is Poisson distributed with an expected value of $\widetilde{G} = \int_{\mathbf{a} \in \mathcal{A}} \mathcal{G}(\mathbf{a}) \, \mathrm{d}\mathbf{a}$ packets per time slot (pps). Given $\mathcal{N}$, the locations of the interferers are i.i.d. with each pdf given by $f_{\mathbf{a}}(\mathbf{a}|\mathcal{N}) = \mathcal{G}(\mathbf{a})/\widetilde{G}, \mathbf{a} \in \mathcal{A}$.

6.2.2.2 Path Loss

In our model, all transmitters use the same power. However, each signal experiences a path loss depending on the propagation distance, a. The local-mean power, i.e., the averaged received power for a given a, follows the path loss law $\bar{\gamma} = a^{-\beta}$. For a typical cellular land mobile radio environment, the path loss exponent lies between $3 \leq \beta \leq 4$. We choose $\beta = 4$, which additionally allows closed-form expression to be obtained for the capture probability [53].

6.2.2.3 Wireless Channel

We assume that all transmitter-receiver links experience flat Rayleigh fading, so the power γ received over a distance a follows an exponential distribution around the local-mean power $\bar{\gamma}$, i.e., $f_\gamma(\gamma) = \exp(-\gamma/\bar{\gamma})/\bar{\gamma}$. We assume that all channels are quasi-static at least over one slot. The channel between S and D, however, may vary during the RTS/CTS cycle:

- Quasi-Static (QS) channel: the SD channel is the same for all slots;
- Quasi-static, non-reciprocal (QSNR) channel: the forward channel from S to D (in the RTS and DATA phases) is independent of the reverse channel from D to S (in the CTS and ACK phases), but both channels are constant;
- Independent and identically distributed (I.I.D.) channel: the SD channel is i.i.d. for all slots.

The QS channel represents a reciprocal channel (i.e., the forward and reverse channels are the same) with a fixed channel fading throughout the RTS/CTS cycle. The QSNR channel reflects a non-reciprocal channel with independent forward and reverse channels, but still quasi-static. The I.I.D. channel is sometimes called the *block fading channel* in the literature [110]. The results obtained for this channel are also relevant for a slow, correlated fading when P is sufficiently large, since it leads to a diverse collection of SINRs in the DATA phase.

6.2.3 Capture Model

The instantaneous SINR is defined as

$$\mathsf{SINR} = \frac{\gamma^{\text{sig}}}{\gamma^{\text{int}}}, \quad \gamma^{\text{int}} = N_{\text{o}} + \sum_{i=1}^{\mathcal{N}} \gamma_i, \tag{6.3}$$

where γ^{sig} is the instantaneous power of the wanted signal, and γ^{int} is the sum of the noise power N_{o} and the instantaneous power of the ith interferer γ_i, where $i = 1, 2, \cdots, \mathcal{N}$. We treat the SINR as a random variable that depends on the random variables $\gamma^{\text{sig}}, \gamma^{\text{int}}$. When the SINR is larger than a fixed *capture ratio*, we assume that the packet captures the destination.

The mutual information between a transmitted packet and the received packet is $\mathcal{I} \triangleq \log_2(1 + \mathsf{SINR})$ using an ideal Gaussian code. According to information-theoretic results for Gaussian channels [111], a packet encoded at data rate R would experience an *information outage* if $\mathcal{I} < R$, or equivalently if

$$\mathsf{SINR} < z(R) \triangleq 2^R - 1. \tag{6.4}$$

Here, $z(R)$ is also the capture ratio. In practice, capture requires higher SINR, but we follow [70] by assuming that a capture occurs if and only if (6.4) is satisfied. This approximation predicts the capture probability accurately within a few dBs of SNR (or SINR in this case) using practical trellis codes [110]. Practical codes can thus be accounted for by adding a margin to $z(R)$.

6.2.4 Capture Model in Different Phases of RTS/CTS Cycle

Let $\gamma_\Phi^{\text{sig}}, \gamma_\Phi^{\text{int}}, R_\Phi$ and $z_\Phi = z(R_\Phi)$ denote, respectively, the signal power, total interference power, rate and capture ratio of a packet in phase $\Phi \in \{\text{R}, \text{C}, \text{D}\}$. The RTS or CTS packet is one slot long. According to (6.4), for the RTS and CTS slots

$$\mathcal{C}_\Phi \Leftrightarrow \frac{\gamma_\Phi^{\text{sig}}}{\gamma_\Phi^{\text{int}}} > z_\Phi, \quad \Phi = \text{R}, \text{C}. \tag{6.5}$$

Next, we consider two models for the capture of a given slot in the DATA phase.

6.2.4.1 Slot-by-Slot DATA Detection

In this mode, each slot in DATA is encoded independently, and hence also detected and recovered independently[2]. We recall that the (ALOHA) interference at different slots is i.i.d. Hence, similar to (6.5), the DATA capture is determined as

$$\mathcal{C}_\text{D} \Leftrightarrow \frac{\gamma_\text{D}^{\text{sig}}}{\gamma_\text{D}^{\text{int}}} > z_\text{D}. \tag{6.6}$$

In this approach, one bit is used to acknowledge each DATA slot, so multiple ACK bits are transmitted in the ACK packet. This approach is similar to the *block-acknowledgement mode* adopted in IEEE 802.11e as an optional feature to reduce the MAC overheads [113], where RTS/CTS is in fact recommended for protection.

6.2.4.2 Entire-Packet DATA Detection

Alternatively, all the data can be encoded as a (long) codeword and transmitted as a packet over P slots. The DATA is thus detected as an entire packet. When the packet length is sufficiently large, an achievable rate in bit/symbol is given by the *averaged mutual information* $\overline{\mathcal{I}}_\text{D} \triangleq \frac{1}{P}\sum \mathcal{I}_{\text{D},k}$ [70], where $\mathcal{I}_{\text{D},k} = \log_2(1 + \mathsf{SINR}_k)$ is the mutual information in slot k of the DATA phase. Further, if a packet is recovered, every DATA slot in it must also be recovered. In accordance we model that

$$\mathcal{C}_\text{D} \Leftrightarrow \overline{\mathcal{I}}_\text{D} > R_\text{D}. \tag{6.7}$$

6.3 System Performance

This section formulates the capture probability and link throughput for the RTS/CTS protocol. We fix P, rather than consider a composite traffic with packets of varying length, to investigate for which P the RTS/CTS option becomes favorable.

[2]This may be appropriate, for instance, in a communication system that uses OFDMA [112], by sending each data stream over a separate subcarrier (instead of a slot). If the subcarriers are spaced sufficiently apart, as compared to the channel coherence bandwidth, then the subcarriers experience i.i.d. channel fading.

6.3.1 DATA Capture Probability

From (6.2), the DATA capture probability is

$$\begin{aligned} \Pr(\mathcal{C}_{\mathrm{D}}) &= \Pr(\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}}, \mathcal{C}_{\mathrm{D}}) \\ &= \Pr(\mathcal{C}_{\mathrm{R}}) \Pr(\mathcal{C}_{\mathrm{C}}|\mathcal{C}_{\mathrm{R}}) \Pr(\mathcal{C}_{\mathrm{D}}|\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}}). \end{aligned} \tag{6.8}$$

We refine [51, 54, 55] by neither assuming that $\Pr(\mathcal{C}_{\mathrm{C}}|\mathcal{C}_{\mathrm{R}}) = 1$ nor $\Pr(\mathcal{C}_{\mathrm{D}}|\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}}) = 1$.

6.3.2 Throughput

Since we assume that RTS/CTS cycles do not overlap, the *cycle time*, i.e., the duration of each RTS/CTS cycle, is i.i.d. Hence, the cycle time forms a renewal process. Let $s_k \in \{0, L_{\mathrm{sym}} \times R_{\mathrm{D}}\}$ be the number of bits recovered in slot k, where L_{sym} is the number of symbols sent per slot. We note that in the RTS and CTS phases $s_k = 0$ since only overhead is sent. By applying the renewal-reward theorem [104], the time average of the throughput in bit/symbol is

$$\begin{aligned} \bar{s}(a_s, R_{\mathrm{R}}, R_{\mathrm{C}}, R_{\mathrm{D}}, P) &\triangleq \lim_{N\to\infty} \frac{1}{L_{\mathrm{sym}}} \frac{1}{N} \sum_{k=1}^{N} s_k \\ &= \frac{1}{L_{\mathrm{sym}}} \frac{\mathbb{E}[\mathcal{R}]}{\mathbb{E}[\mathcal{T}]} \end{aligned} \tag{6.9}$$

with probability one. Here, the expectation $\mathbb{E}$ is defined with respect to the events in one cycle; $\mathcal{R}$ is the *reward* in the form of the number of bits recovered in one cycle, and $\mathcal{T}$ is the cycle time in slots. The argument $a_s \triangleq |\mathbf{a}_{\mathsf{S}} - \mathbf{a}_{\mathsf{D}}|$ is defined as the distance between S and D. As explicitly indicated above, $\bar{s}$ depends on many parameters, but we take $a_s, R_{\mathrm{R}}, R_{\mathrm{C}}$ to be fixed (and drop these arguments subsequently). We now analyze how the DATA rate R_{D} and DATA length P affect the throughput.

Let us denote the complement of $\mathcal{C}$ as $\mathcal{E}$, i.e., $\mathcal{E}$ is the event that a slot is received in error. There are only two possible situations, S1 and S2, when a cycle terminates[3] from the perspective of S:

S1: D does not recover the RTS *or* S does not recover the CTS, i.e., $\mathcal{E}_{\mathrm{R}} \cup (\mathcal{C}_{\mathrm{R}} \cap \mathcal{E}_{\mathrm{C}})$ occurs. In either case, S has to wait for the CTS to arrive before attempting to recover it, so a cycle time of 2 time slots is always consumed. S1 occurs with probability $\Pr(\mathcal{E}_{\mathrm{R}}) + \Pr(\mathcal{C}_{\mathrm{R}}, \mathcal{E}_{\mathrm{C}}) = 1 - \Pr(\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}})$.

S2: S recovers the CTS, i.e., $\mathcal{C}_{\mathrm{R}} \cap \mathcal{C}_{\mathrm{C}}$. Hence, the DATA will be sent and a cycle time of $P + 2$ slots will be consumed. S2 occurs with probability $\Pr(\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}})$.

[3] The definition of the end of a cycle from the perspective of D can be different. For example, if D does not recover a RTS, the cycle is not even initiated as seen from D's perspective, while S has to wait for the CTS phase to be over (even without recovering any CTS) before it is sure that the cycle has terminated.

On average, $P\Pr(\mathcal{C}_\mathrm{D})$ slots are successfully transported from S to D. Hence, the expected reward (in bits per cycle) is $\mathbb{E}[\mathcal{R}] = L_\mathrm{sym} R_\mathrm{D} P \Pr(\mathcal{C}_\mathrm{D})$. Taking into account that situations S1 and S2 use 2 and $P+2$ slots, respectively, the expected cycle time (in slots) evaluates as $\mathbb{E}[\mathcal{T}] = 2 + P\Pr(\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C})$. Using (6.8) and the above results for $\mathbb{E}[\mathcal{R}]$ and $\mathbb{E}[\mathcal{T}]$, the throughput (6.9) may be expressed as

$$\bar{s}(R_\mathrm{D}, P) = \eta R_\mathrm{D} \Pr(\mathcal{C}_\mathrm{D}|\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C}; R_\mathrm{D}, P). \tag{6.10}$$

where

$$\eta \triangleq \frac{P\Pr(\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C})}{2 + P\Pr(\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C})}.$$

For clarity, R_D and P are denoted explicitly as parameters in $\Pr(\mathcal{C}_\mathrm{D}|\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C}; R_\mathrm{D}, P)$. The fractional protocol overhead η increases monotonically and asymptotically approaches

$$\lim_{P\to\infty} \eta = 1$$

for any $\Pr(\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C}) > 0$. Hence, the asymptotic throughput for large P is given by

$$\bar{\bar{s}}(R_\mathrm{D}) \triangleq \lim_{P\to\infty} \bar{s}(R_\mathrm{D}, P) \tag{6.11}$$

$$= R_\mathrm{D} \lim_{P\to\infty} \Pr(\mathcal{C}_\mathrm{D}|\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C}; R_\mathrm{D}, P). \tag{6.12}$$

In particular for slot-by-slot detection, since the slots are decoded independently the DATA capture probability does not depend on P, so the asymptotic throughput becomes $\bar{\bar{s}}(R_\mathrm{D}) = R_\mathrm{D} \Pr(\mathcal{C}_\mathrm{D}|\mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C}; R_\mathrm{D})$.

6.4 Capture Probabilities: Detailed Analysis

In Section 6.4.1, we model the traffic in the various phases of the RTS/CTS cycle as non-homogeneous Poisson processes. Then, Section 6.4.2 derives the general expressions of the capture probabilities in different phases. Section 6.4.3 relates these results to compute the desired conditional capture probabilities, eventually leading to the throughput (6.10). The analysis is conducted for the I.I.D. channel, but is also relevant for the QS and QSNR channels, as discussed in Section 6.4.4.

6.4.1 Traffic in Different Phases

We are interested in the traffic conditioned on a successive-capture event $\mathcal{C}$ in the set $\mathcal{S}_\mathcal{C} \triangleq \{\emptyset, \mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{C}, \mathcal{C}_\mathrm{D}\}$, or equivalently $\mathcal{S}_\mathcal{C} = \{\emptyset, \mathcal{C}_\mathrm{R}, \mathcal{C}_\mathrm{R} \cap \mathcal{C}_\mathrm{C}, \mathcal{C}_\mathrm{R} \cap \mathcal{C}_\mathrm{C} \cap \mathcal{C}_\mathrm{D}\}$. Here, the non-event $\emptyset$ denotes that the protocol is just starting. For every phase $\Phi \in \{\mathrm{R}, \mathrm{C}, \mathrm{D}\}$, we denote

- $\mathcal{G}_\Phi(\mathbf{a}|\mathcal{C})$ as the conditional traffic intensity in phase Φ (i.e., when the Φ packet is sent), and

- $\mathcal{P}_\Phi(\mathbf{a}|\mathcal{C})$ as the conditional probability that an receiver at $\mathbf{a}$ recovers the Φ packet. This receiver can be an interferer, S or D.

Both of the above functions give a quantitative picture of the network in the Φ phase: $\mathcal{G}_\Phi(\mathbf{a}|\mathcal{C})$ indicates the likelihood of having interference that originates from $\mathbf{a}$, while $\mathcal{P}_\Phi(\mathbf{a}|\mathcal{C})$ indicates the likelihood of receiving a packet at $\mathbf{a}$. Our expressions imply a refined notion of hidden and exposed nodes, which we study from a probabilistic view. Specifically, a potentially harmful interferer remains uninhibited with probability $1-\mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C}_\mathrm{R})$ in the CTS phase and with probability $(1-\mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C}_\mathrm{R},\mathcal{C}_\mathrm{C}))(1-\mathcal{P}_\mathrm{C}(\mathbf{a}|\mathcal{C}_\mathrm{R},\mathcal{C}_\mathrm{C}))$ in the DATA phase, while a harmless remote node is unjustly inhibited with probability $\mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C}_\mathrm{R})$ in the CTS phase and with probability $1-(1-\mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C}_\mathrm{R},\mathcal{C}_\mathrm{C}))(1-\mathcal{P}_\mathrm{C}(\mathbf{a}|\mathcal{C}_\mathrm{R},\mathcal{C}_\mathrm{C}))$ in the DATA phase.

Thus, the quantity $\mathcal{P}_\Phi(\mathbf{a}|\mathcal{C})$ is particularly interesting for cases when $\mathcal{C}$ denotes an event in the present and prior phases of the protocol (e.g., $\Phi=\mathrm{C}$ and $\mathcal{C}=\mathcal{C}_\mathrm{R}\cap\mathcal{C}_\mathrm{C}$) and $\mathbf{a}$ refers to a potential interferer. We do not include degenerate cases of $\mathcal{P}_\Phi(\mathbf{a}|\mathcal{C})=1$ when the capture event under consideration is the same or a subset of the conditioning event.

6.4.1.1 Distribution of $\mathbb{T}_\mathrm{C}$

In the CTS phase, the interferers transmit if they have data to transmit (which are always independent events) *and* if they have not captured the RTS. Given that the RTS has captured D, i.e., $\mathcal{C}_\mathrm{R}$ has occurred, it is likely that the interference around D is small in the RTS phase. These interferers would also have captured the RTS with high probability and thus not likely to transmit in the CTS phase. Hence, the probability that a particular interferer near D transmits is always close to zero given $\mathcal{C}_\mathrm{R}$, regardless of whether another interferer transmits in the CTS phase. This means that the interferers transmit almost independently in the CTS phase as long as one interferer is near D. This independence applies well for interferers in an area around D, where the area depends on the RTS rate. Since the RTS rate is typically set fairly low, this area of independent interferers is typically large. For interferers outside this large area that are fairly far away from D, their transmission may not be independent, but their effect on the reception of D may be negligible.

These heuristic arguments suggest that modeling the transmissions of the interferers as independent in the CTS phase is accurate; the same arguments also apply for the DATA phase. Moreover, simulation results lend support to this model, see Fig. 6.6. Henceforth, to simplify the analysis, we assume that the interferers transmit independently in the CTS and DATA phases.

The RTS captures an interferer located at $\mathbf{a}$ with probability $1-\mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C}_\mathrm{R})$. Taking the transmissions of the interferers as independent, we can apply the thinning property of the spatial Poisson process [114]. It follows that the traffic in the CTS phase is (i) a Poisson process and (ii) the traffic intensity is $1-\mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C}_\mathrm{R})$ multiplied by the original traffic intensity if there were no inhibition (i.e., G_o). So, $\mathbb{T}_\mathrm{C}$ conditioned on

$\mathcal{C}_\mathrm{R}$ is described by a non-homogeneous Poisson process with traffic intensity

$$\mathcal{G}_\mathrm{C}(\mathbf{a}|\mathcal{C}) = G_o\left(1 - \mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C})\right), \quad \mathcal{C} = \mathcal{C}_\mathrm{R}. \tag{6.13}$$

This illustrates how an inhibition affects the traffic intensity in the network over time.

6.4.1.2 Distribution of $\mathbb{T}_\mathrm{D}$

In the DATA phase, the interferer is *not* inhibited if it neither recovers the RTS nor the CTS. Given $\mathcal{C} = \mathcal{C}_\mathrm{R} \cap \mathcal{C}_\mathrm{C}$, this occurs with probability $\left(1 - \mathcal{P}_\mathrm{C}(\mathbf{a}|\mathcal{C})\right)\left(1 - \mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C})\right)$. Similarly, by taking the inhibitions as independent and using the thinning property, $\mathbb{T}_\mathrm{D}$ conditioned on $\mathcal{C}_\mathrm{R} \cap \mathcal{C}_\mathrm{C}$ is described by a non-homogeneous Poisson process with traffic intensity

$$\mathcal{G}_\mathrm{D}(\mathbf{a}|\mathcal{C}) = G_o\left(1 - \mathcal{P}_\mathrm{C}(\mathbf{a}|\mathcal{C})\right)\left(1 - \mathcal{P}_\mathrm{R}(\mathbf{a}|\mathcal{C})\right), \tag{6.14}$$

where $\mathcal{C} = \mathcal{C}_\mathrm{R} \cap \mathcal{C}_\mathrm{C}$.

6.4.1.3 Generalization

Generally, we model the traffic conditioned on any $\mathcal{C} \in \mathcal{S}_\mathcal{C}$ by Poisson processes. In particular, the traffic in the CTS and DATA phase have traffic intensity (6.13) and (6.14), respectively, by replacing $\mathcal{C}$ accordingly, while in the RTS phase the traffic intensity $\mathcal{G}_\mathrm{R}(\mathbf{a}|\mathcal{C})$ will be determined subsequently.

6.4.2 Capture Probability

We consider a node at $\mathbf{a}_\mathrm{tx}$ that transmits a packet to a receiver at $\mathbf{a}_\mathrm{rx}$ at rate $R = 2^z - 1$. The traffic during transmission follows a Poisson process with traffic intensity $\mathcal{G}(\mathbf{a}), \mathbf{a} \in \mathcal{A}$. The capture event $\mathcal{C}_\mathrm{ALO}$ is independent of past capture events (e.g., in the ALOHA protocol), but depends on $\mathcal{G}$. Its capture probability is given by [53]

$$\begin{aligned} & \Pr\left\{\mathcal{C}_\mathrm{ALO}|\mathbf{a}_\mathrm{tx}, \mathbf{a}_\mathrm{rx}, z, \mathcal{G}\right\} \\ = & \int_\mathbb{T} \Pr\left\{\mathcal{C}_\mathrm{ALO}|\mathbf{a}_\mathrm{tx}, \mathbf{a}_\mathrm{rx}, z, \mathbb{T}\right\} f_\mathbb{T}(\mathbb{T}|\mathcal{G})\,\mathrm{d}\mathbb{T} && (6.15) \\ = & \exp\left(-\frac{zN_\mathrm{o}}{\bar{\gamma}} - \int_{\mathbf{a}\in\mathcal{A}} \mathcal{J}(\mathbf{a})\,\mathrm{d}\mathbf{a}\right), && (6.16) \end{aligned}$$

where $\mathcal{J}(\mathbf{a}) \triangleq W(z, |\mathbf{a}_\mathrm{tx} - \mathbf{a}_\mathrm{rx}|, |\mathbf{a} - \mathbf{a}_\mathrm{rx}|)\mathcal{G}(\mathbf{a})$ and the *vulnerability weight factor* is

$$W(z, x, y) \triangleq zx^\beta/(zx^\beta + y^\beta). \tag{6.17}$$

For the derivation of (6.16) we refer to [53] for the case of ALOHA.

$$\xrightarrow{(6.27)} \mathcal{G}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}}) \xrightarrow[\mathcal{C}=\mathcal{C}_{\mathrm{R}}]{(6.22)} \mathcal{P}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}}) \xrightarrow{(6.13)} \mathcal{G}_{\mathrm{C}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}}) \xrightarrow{(6.26)} \Pr(\mathcal{C}_{\mathrm{C}}|\mathcal{C}_{\mathrm{R}})$$

Fig. 6.3: Relationship of capture probabilities and traffic intensities for calculating $\Pr(\mathcal{C}_{\mathrm{C}}|\mathcal{C}_{\mathrm{R}})$.

Next, we make use of (6.15) to find a generic expression for $\mathcal{P}_{\Phi}(\mathbf{a}|\mathcal{C}), \Phi = \mathrm{R}, \mathrm{C}, \mathrm{D}$. We will exploit the observation that given the traffic $\mathbb{T}_{\Phi}$, the capture probability of a receiver in phase Φ is independent of any event $\mathcal{C}$, i.e.,

$$\mathcal{P}_{\Phi}(\mathbf{a}|\mathcal{C}, \mathbb{T}_{\Phi}) = \mathcal{P}_{\Phi}(\mathbf{a}|\mathbb{T}_{\Phi}). \tag{6.18}$$

This is because $\mathbb{T}_{\Phi}$ is sufficient to determine the capture probability, by using (6.5) for $\Phi = \mathrm{R}$ and $\Phi = \mathrm{C}$, or by using (6.6) or (6.7) for $\Phi = \mathrm{D}$.

As a shorthand, we denote $\mathcal{G}_{\Phi}(\mathbf{a}|\mathcal{C}), \mathbf{a} \in \mathcal{A}$, as $\mathcal{G}_{\Phi|\mathcal{C}}$. Given $\mathcal{C}$, the probability of capturing the Φ packet at $\mathbf{a}$ is

$$\mathcal{P}_{\Phi}(\mathbf{a}|\mathcal{C}) = \int_{\mathbb{T}_{\Phi}} \mathcal{P}_{\Phi}(\mathbf{a}|\mathcal{C}, \mathbb{T}_{\Phi}) f_{\mathbb{T}_{\Phi}}(\mathbb{T}_{\Phi}|\mathcal{G}_{\Phi|\mathcal{C}})\, \mathrm{d}\mathbb{T}_{\Phi} \tag{6.19}$$

$$= \int_{\mathbb{T}_{\Phi}} \mathcal{P}_{\Phi}(\mathbf{a}|\mathbb{T}_{\Phi}) f_{\mathbb{T}_{\Phi}}(\mathbb{T}_{\Phi}|\mathcal{G}_{\Phi|\mathcal{C}})\, \mathrm{d}\mathbb{T}_{\Phi} \tag{6.20}$$

$$= \int_{\mathbb{T}_{\Phi}} \Pr\left\{\mathcal{C}_{\mathrm{ALO}}|\mathbf{a}_{\mathrm{tx}}, \mathbf{a}, z_{\Phi}, \mathbb{T}_{\Phi}\right\} f_{\mathbb{T}_{\Phi}}(\mathbb{T}_{\Phi}|\mathcal{G}_{\Phi|\mathcal{C}})\, \mathrm{d}\mathbb{T}_{\Phi}, \tag{6.21}$$

where $\mathbf{a}_{\mathrm{tx}} = \mathbf{a}_{\mathsf{S}}$ for $\Phi = \mathrm{R}, \mathrm{D}$ and $\mathbf{a}_{\mathrm{tx}} = \mathbf{a}_{\mathsf{S}}$ for $\Phi = \mathrm{C}$. Here, (6.19) follows from the law of total probability and (6.20) follows from (6.18), (6.21) follows from the definition of $\mathcal{P}_{\Phi}(\mathbf{a}|\mathbb{T}_{\Phi})$. Comparing (6.21) with (6.15), we get

$$\mathcal{P}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}) = \Pr\left\{\mathcal{C}_{\mathrm{ALO}}|\mathbf{a}_{\mathsf{S}}, \mathbf{a}, z_{\mathrm{R}}, \mathcal{G}_{\mathrm{R}|\mathcal{C}}\right\}, \tag{6.22}$$

$$\mathcal{P}_{\mathrm{C}}(\mathbf{a}|\mathcal{C}) = \Pr\left\{\mathcal{C}_{\mathrm{ALO}}|\mathbf{a}_{\mathsf{D}}, \mathbf{a}, z_{\mathrm{C}}, \mathcal{G}_{\mathrm{C}|\mathcal{C}}\right\}, \tag{6.23}$$

$$\mathcal{P}_{\mathrm{D}}(\mathbf{a}|\mathcal{C}) = \Pr\left\{\mathcal{C}_{\mathrm{ALO}}|\mathbf{a}_{\mathsf{S}}, \mathbf{a}, z_{\mathrm{D}}, \mathcal{G}_{\mathrm{D}|\mathcal{C}}\right\}. \tag{6.24}$$

These probabilities translate (6.15) to the results for the RTS/CTS protocol. Interestingly, the conditioning on the capture probability becomes a conditioning on the traffic intensity. We emphasize that the receiver considered in (6.22), with arbitrary location $\mathbf{a}$, can be an interferer, S or D.

6.4.3 Relating Traffic Intensity and Capture Probability

We now apply (6.22) to obtain $\Pr(\mathcal{C}_{\mathrm{R}})$, $\Pr(\mathcal{C}_{\mathrm{C}}|\mathcal{C}_{\mathrm{R}})$ and $\Pr(\mathcal{C}_{\mathrm{D}}|\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}})$ in the DATA capture probability (6.8).

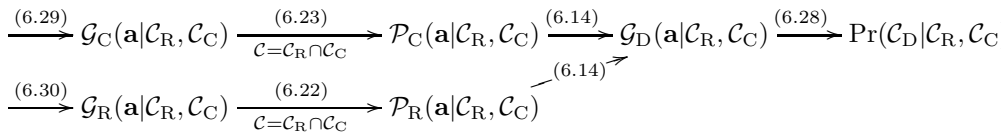


Fig. 6.4: Relationship of capture probabilities and traffic intensities over time and space for calculating $\Pr(\mathcal{C}_{\mathrm{D}}|\mathcal{C}_{\mathrm{R}},\mathcal{C}_{\mathrm{C}})$.

6.4.3.1 Capture Probability of RTS

In the RTS phase no inhibition is active yet, so (6.15) applies, in which case the receiver is S, the transmitter ia D, $z = z_{\mathrm{R}}$ and $\mathcal{G}(\mathbf{a}) = G_o, \mathbf{a} \in \mathcal{A}$:

$$\Pr(\mathcal{C}_{\mathrm{R}}) = \exp\left(-z_{\mathrm{R}} N_{\mathrm{o}}/\bar{\gamma} - a_s^2 \pi^2 G_o \sqrt{z_{\mathrm{R}}}/2\right), \tag{6.25}$$

with $a_s = |\mathbf{a}_{\mathsf{S}} - \mathbf{a}_{\mathsf{D}}|$. This closed-form expression corresponds to the results derived in [53].

6.4.3.2 Capture Probability of CTS

To find $\Pr(\mathcal{C}_{\mathrm{C}}|\mathcal{C}_{\mathrm{R}})$, we apply (6.23) by letting the receiver be S (so $\mathbf{a} = \mathbf{a}_{\mathsf{S}}$) and $\mathcal{C} = \mathcal{C}_{\mathrm{R}}$:

$$\Pr(\mathcal{C}_{\mathrm{C}}|\mathcal{C}_{\mathrm{R}}) = \Pr\left\{\mathcal{C}_{\mathrm{ALO}}|\mathbf{a}_{\mathsf{D}}, \mathbf{a}_{\mathsf{S}}, z_{\mathrm{C}}, \mathcal{G}_{\mathrm{C}|\mathcal{C}_{\mathrm{R}}}\right\}. \tag{6.26}$$

Fig. 6.3 summarizes the subsequent calculations involved. To calculate $\mathcal{G}_{\mathrm{C}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}})$ in (6.26), we use (6.13), which in turn requires $\mathcal{P}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}})$, $\mathbf{a} \in \mathcal{A}$. To compute $\mathcal{P}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}})$, we apply (6.22) by letting the receiver be an interferer at $\mathbf{a}$, and letting $\mathcal{C} = \mathcal{C}_{\mathrm{R}}$. Finally, we require $\mathcal{G}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}})$ to be known. As derived in Appendix 6.A, $\mathcal{G}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}})$ can be expressed as

$$\mathcal{G}_{\mathrm{R}}(\mathbf{a}|\mathcal{C}_{\mathrm{R}}) = G_o(1 - W(z_{\mathrm{R}}, a_s, |\mathbf{a} - \mathbf{a}_{\mathsf{D}}|)). \tag{6.27}$$

This conditional traffic intensity can be interpreted as an *a posteriori* traffic intensity, knowing $\mathcal{C}_{\mathrm{R}}$. It differs from the *a priori* traffic intensity $\mathcal{G}_{\mathrm{R}}(\mathbf{a}) = G_o$. Intuitively, we expect that given $\mathcal{C}_{\mathrm{R}}$ the amount of interference that occurred around D in the RTS phase *was* likely to be small – otherwise the RTS would not have captured D. From W defined in (6.17), we see indeed that the traffic intensity (6.27) is small around D and approaches G_o further away from D.

6.4.3.3 Capture Probability of DATA

We consider the DATA capture probability $\Pr(\mathcal{C}_{\mathrm{D}}|\mathcal{C}_{\mathrm{R}},\mathcal{C}_{\mathrm{C}})$ using either slot-by-slot or entire-packet DATA detection of Section 6.2.4. We note that both approaches of detection are equivalent for unit DATA packet length, i.e., $P = 1$.

For slot-by-slot packet detection, we apply (6.24) by letting the receiver be D (so $\mathbf{a} = \mathbf{a}_{\mathsf{D}}$) and $\mathcal{C} = \mathcal{C}_{\rm R} \cap \mathcal{C}_{\rm C}$. We obtain

$$\Pr(\mathcal{C}_{\rm D}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}) = \Pr\left\{\mathcal{C}_{\rm ALO}|\mathbf{a}_{\mathsf{S}}, \mathbf{a}_{\mathsf{D}}, z_{\rm D}, \mathcal{G}_{{\rm D}|\mathcal{C}_{\rm R},\mathcal{C}_{\rm C}}\right\}, \tag{6.28}$$

where the traffic intensity $\mathcal{G}_{\rm D}(\mathbf{a}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C})$ has been shown to be given by (6.14). Following the same arguments as earlier, we arrive at a sequence of computations summarized in Fig. 6.4. Finally, what is left to compute are the *a posteriori* traffic intensities

$$\begin{aligned} \mathcal{G}_{\rm C}(\mathbf{a}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}) &= (1 - W(z_{\rm C}, a_s, |\mathbf{a} - \mathbf{a}_{\mathsf{S}}|))\mathcal{G}_{\rm C}(\mathbf{a}|\mathcal{C}_{\rm R}), & (6.29)\\ \mathcal{G}_{\rm R}(\mathbf{a}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}) &= (1 - W(z_{\rm C}, a_s, |\mathbf{a} - \mathbf{a}_{\mathsf{S}}|))\mathcal{G}_{\rm R}(\mathbf{a}|\mathcal{C}_{\rm R}), & (6.30) \end{aligned}$$

where the derivations are found in Appendix 6.B. Here, $\mathcal{G}_{\rm C}(\mathbf{a}|\mathcal{C}_{\rm R})$ and $\mathcal{G}_{\rm R}(\mathbf{a}|\mathcal{C}_{\rm R})$ are available from (6.13) and (6.27), respectively.

For entire-packet detection of DATA, the capture of the packet, or equivalently of a slot in the packet, is governed by (6.7). For I.I.D. channels, the instantaneous SINR in (6.3) is i.i.d., and thus so is the mutual information $\mathcal{I}_{\rm D}$ (we drop the time index k). Hence, as the packet length P increases, the distribution of the averaged mutual information $\overline{\mathcal{I}}_{\rm D}$ converges to a Gaussian distribution with mean and variance given respectively by

$$\begin{aligned} \mu &= \mathbb{E}[\mathcal{I}_{{\rm D},k}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}],\\ \sigma^2 &= \left(\mathbb{E}\left[\mathcal{I}^2_{{\rm D},k}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}\right] - \mu^2\right)/P. \end{aligned}$$

For our RTS/CTS protocol, we have validated in simulations that this Gaussian approximation is fairly accurate even for small P, say $P = 2$ (also see Fig. 6.8). The conditional pdf of $\mathcal{I}_{\rm D}$ can be calculated from (6.28) which can be written alternatively as $\Pr(\mathcal{I}_{\rm D} > 2^{z_{\rm D}} - 1|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C})$. Hence, μ and σ^2 can also be computed. The capture probability is then approximated as

$$\Pr(\mathcal{C}_{\rm D}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}) \approx Q\left(\frac{R_{\rm D} - \mu}{\sigma}\right), \tag{6.31}$$

where $Q(x) = \int_x^\infty \exp(-y^2/2)/\sqrt{2\pi}{\rm d}y$ is the Gaussian error integral function. In particular, consider the case that $P \to \infty$. By the law of large number $\overline{\mathcal{I}}_{\rm D}$ approaches μ, a constant. Hence, a DATA packet is recovered if and only if its rate $R_{\rm D}$ is less than $\overline{\mathcal{I}}_{\rm D}$. It follows that the throughput (6.12) increases linearly with $R_{\rm D}$ for $R_{\rm D} < \mu$, but abruptly drops to 0 for $R_{\rm D} \geq \mu$.

Some of the conditional probabilities and traffic intensities depicted in Fig. 6.3 and Fig. 6.4 do not yield closed-form solutions. However, numerical results have been obtained. Intermediate results are presented in this chapter in the form of contour plots of $\mathcal{G}_{\rm C}(\mathbf{a}|\mathcal{C}_{\rm R}), \mathcal{G}_{\rm D}(\mathbf{a}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C})$ in Fig. 6.5 indicating the extent of the RTS and CTS reservations.

6.4.4 DATA Capture Probabilities in Different Channels

We consider slot-by-slot DATA packet detection. The *conditional* DATA capture probability given $\mathbb{T}_{\text{cycle}}$ for the QS, QSNR and I.I.D. channels are related as follows:

$$\begin{aligned} \Pr(\mathcal{C}_{\text{D}}|\mathbb{T}_{\text{cycle}};\ \text{I.I.D.}) &\leq \Pr(\mathcal{C}_{\text{D}}|\mathbb{T}_{\text{cycle}};\ \text{QSNR}) \\ &\leq \Pr(\mathcal{C}_{\text{D}}|\mathbb{T}_{\text{cycle}};\ \text{QS}). \end{aligned} \tag{6.32}$$

Further, this relationship holds *without* conditioning on the traffic:

$$\Pr(\mathcal{C}_{\text{D}};\ \text{I.I.D.}) \leq \Pr(\mathcal{C}_{\text{D}};\ \text{QSNR}) \leq \Pr(\mathcal{C}_{\text{D}};\ \text{QS}). \tag{6.33}$$

The proofs for (6.32) and (6.33) are given in Appendices 6.C and 6.D, respectively. Although we cannot find analytical solutions for the cycle capture probabilities for the QS and QSNR channels, an analysis for the I.I.D. channel provides a lower bound on the capture probabilities. For comparison, results for the other channels are obtained by simulations.

For entire-packet DATA detection, on the other hand, an inequality such as (6.33) cannot be established; simulation results confirm that the relationship of the capture probabilities depend generally on P (see Figure 6.8).

6.5 Numerical Results

For the I.I.D. channels, we present numerical evaluations of the analytical expressions in Section 6.3. Monte Carlo simulations are conducted for other types of channels. We considered the noise-free case $N_{\text{o}} = 0$ to isolate the effect of interference and we used the following system parameters: $\mathbf{a}_{\mathsf{D}} = [0, 0], \mathbf{a}_{\mathsf{S}} = [0.5, 0], G_o = 1/\pi$ ppsa. In our simulations, to approximate an infinitely large area for the operation region $\mathcal{A}$, we used a square of length 20 centered at the origin. Numerical results show that further increase in the area does not lead to a noticeable change in the computed throughput. The RTS and CTS rates are set to $R_{\text{R}} = R_{\text{C}} = 1$ bit/symbol, corresponding for instance to using QPSK modulation with a code rate of 1/2.

6.5.1 Traffic Intensity

As no spatial reservation is in place before the RTS, the starting traffic intensity is uniform with $G_o = 1/\pi$. Fig. 6.5 shows the normalized traffic intensity as the protocol progresses through the CTS and DATA phases. The traffic during the CTS period $\mathcal{G}_{\text{C}}(\mathbf{a}|\mathcal{C}_{\text{R}})$ reduces mostly in the vicinity of S. This reduction arises because the RTS performs a spatial reservation around S, though with fuzzy boundaries. As a result, the CTS capture probability improves. After the RTS is recovered, D transmits a CTS which reserves the channel during the DATA period. This second reservation is centered around D, but jointly with the first reservation of the RTS, creates an

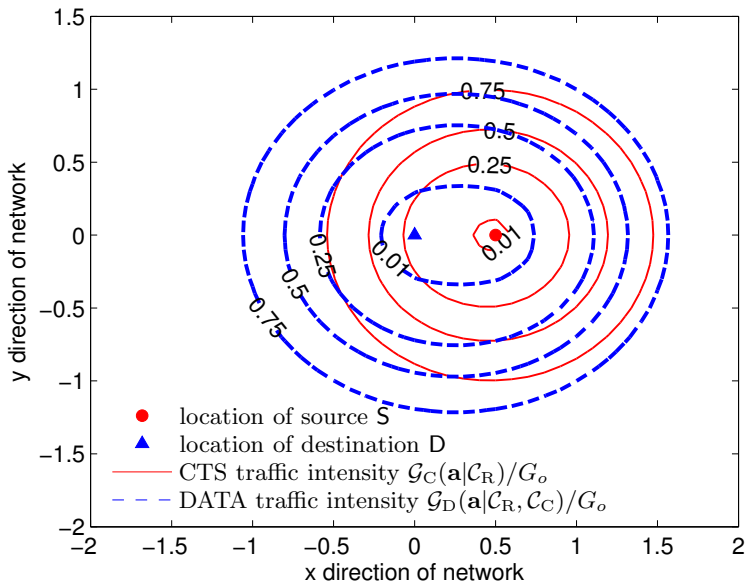


Fig. 6.5: Contour plots of normalized traffic intensities in the CTS and DATA phases. The contours illustrate the spatial effects of inhibition by the RTS and CTS packets. In the CTS phase, only the RTS creates a spatial reservation around S. In the DATA phase, both the RTS and CTS are used, which creates a larger reservation area around S and D.

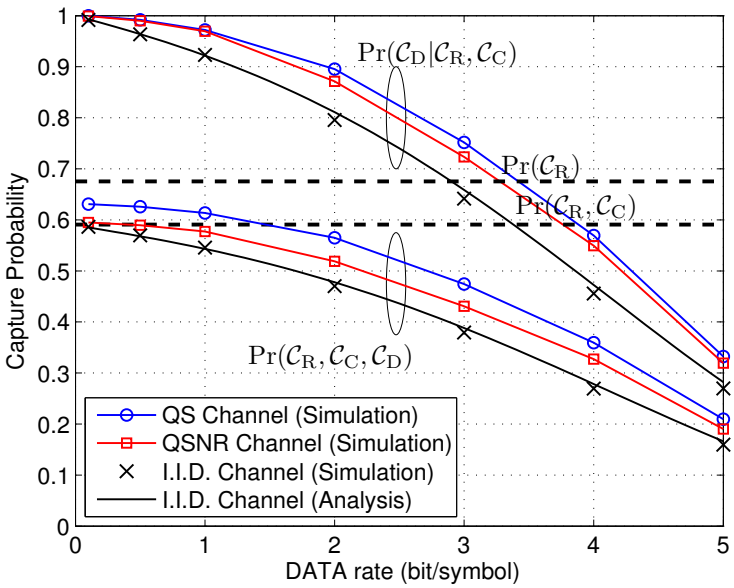


Fig. 6.6: The conditional DATA capture probability $\Pr(\mathcal{C}_D|\mathcal{C}_R,\mathcal{C}_C)$ and DATA capture probability $\Pr(\mathcal{C}_R,\mathcal{C}_C,\mathcal{C}_D)$ for different channels. The RTS capture probability $\Pr(\mathcal{C}_R)$ and the probability that the RTS and CTS are both captured, $\Pr(\mathcal{C}_R,\mathcal{C}_C)$, are also plotted. Here, $P = 1$ is used.

enlarged, overall region of reservation as indicated by $\mathcal{G}_D(\mathbf{a}|\mathcal{C}_R,\mathcal{C}_C)$. Hence, the DATA capture probability also improves.

The use of RTS for inhibition in the DATA phase may be inappropriate, since the inhibition is centered around S rather than D which acts as the receiver in the DATA phase. Alternatively, the RTS inhibition can be eliminated by reducing its inhibition window; this has been considered separately in [115]. Whether it is preferable to have a reduced or increased inhibition window is a tradeoff, in particular, between the DATA capture probability for the users of the RTS/CTS protocol, and the penalty for inhibiting users of the ALOHA protocol.

6.5.2 Capture Probability

The cycle capture probability $\Pr(\mathcal{C}_D) = \Pr(\mathcal{C}_R,\mathcal{C}_C,\mathcal{C}_D)$ and the conditional DATA capture probability $\Pr(\mathcal{C}_D|\mathcal{C}_R,\mathcal{C}_C)$ have been plotted in Fig. 6.6 against the DATA rate R_D. We use $P = 1$ and so the capture probabilities are the same for both slot-by-slot and entire-packet DATA detection. Monte Carlo simulations have been carried out for the I.I.D., QSNR and QS channels using 10,000 RTS/CTS cycles. We observe that the simulation results match the analytical results for the I.I.D. channels,

validating the accuracy of modeling the CTS and PAY traffic as Poisson processes. Next, we observe that the capture probabilities for the QS and QSNR channels are larger than the I.I.D. channel, as predicted by the analysis. For example, at a rate of $R_{\rm D} = 2$, the QS channel is about 20% higher in $\Pr(\mathcal{C}_{\rm D})$ than the I.I.D. channel.

Fig. 6.6 also serves to compare our results with those resulting from the commonly made assumption that $\Pr(\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}, \mathcal{C}_{\rm D}) \approx \Pr(\mathcal{C}_{\rm R})$. For the I.I.D. channel, for instance, we see that $\Pr(\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}, \mathcal{C}_{\rm D}) \ll \Pr(\mathcal{C}_{\rm R}) = 0.68$ for sufficiently large DATA rates. So we conclude that $\Pr(\mathcal{C}_{\rm R})$ (indicated by a dotted line) is overly optimistic. Even $\Pr(\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}) = 0.59$ is optimistic for the cycle capture probability, particularly at high DATA rates, say larger than 2 bit/symbol. These observations justify our effort to explicitly model the probabilistic effects of successive captures and to determine their probabilities.

6.5.3 Throughput

To illustrate the potential of rate optimization, we take $R_{\rm D}$ to be continuous. In practice, rates are discrete. We optimize $R_{\rm D}$ to maximize the link throughput for a fixed RTS and CTS rates, and compare this with slotted ALOHA transmissions, thus without RTS/CTS. For ALOHA transmission at rate $R_{\rm ALOHA}$, the traffic intensity is always G_o. Hence the ALOHA capture probability is found from (6.25) by replacing $z_{\rm R}$ with $2^{R_{\rm ALOHA}} - 1$, and the throughput follows from multiplying $R_{\rm ALOHA}$ with this ALOHA capture probability. For simplicity, we focus on the I.I.D. and QS channels.

6.5.3.1 Slot-by-Slot DATA Detection

We first consider the throughput using slot-by-slot DATA detection. Fig. 6.7 shows the throughput obtained by substituting the simulated or analytical capture probabilities into the analytical expression (6.10). We observe that the throughput in the QS channel is larger than in the I.I.D. channel, which is expected because the DATA capture probability in the QS channel has been shown to be larger. For fixed P and increasing $R_{\rm D}$, $\bar{s}$ increases initially, because transmitting at a (too) low $R_{\rm D}$ limits the throughput. At sufficiently high $R_{\rm D}$, the throughput starts to reduce, due to reduced robustness against interference. We observe that the same optimum DATA rate maximizes the throughout for any P. This is because the DATA capture probability is independent of P for slot-by-slot DATA detection.

Fig. 6.7 shows that for sufficiently large P the RTS/CTS protocol outperforms ALOHA, at any common DATA rate. For moderate P, say $P = 10$, if $R_{\rm D} < 0.75$, the ALOHA throughput can marginally outperform that of the RTS/CTS; but if $R_{\rm D} \approx 3$, substantial throughput gain can be realized. Finally, if P is too small, say $P = 2$, then the overhead of the RTS/CTS protocol is too high, and even with properly tuned DATA rate, the throughput is comparable or less than using the ALOHA protocol.

6.5.3.2 Entire-Packet DATA Detection

Next we consider the throughput using entire-packet DATA detection. From Fig. 6.8, we observe that the simulation results for the I.I.D. channel match the analysis fairly well, despite the approximations used in (6.31). For $P = 1, 2$, we observe that the throughput for the QS channel is larger than for the I.I.D. channel, but as P increases, the throughput for the I.I.D. channel becomes larger than for the QS channel[4]. In particular, Fig. 6.8 shows a distinct sharp optimum when $P \to \infty$, as the law of large numbers on the averaged mutual information takes effect. Hence, the entire packet is recovered if and only if $R_{\mathrm{D}} \leq \mu \approx 3.8$ bit/symbol. Similar to slot-by-slot DATA detection, for moderate or higher P, using the RTS/CTS protocol realizes a higher throughput by transmitting the DATA at around three times the RTS and CTS rates.

Generally, entire-packet DATA detection outperforms slot-by-slot DATA detection, especially for large P. When $P \to \infty$ and for I.I.D. channel, the maximum throughput is close to two times higher. However, for the QS channel, the maximum throughput of using entire-packet DATA detection is only marginally better. We conclude that in an I.D.D. channel entire-packet DATA detection can be used to significantly improve the throughput, but in a QS channel this detection may not be worthwhile, particularly as a substantially higher delay is incurred when decoding the packet as a whole.

Finally, Fig. 6.9 considers how the maximum throughput of the RTS/CTS and ALOHA protocols behave with distance a_s, using independently optimized DATA rates. We consider $P \to \infty$ and entire-packet DATA detection in an I.I.D. channel for the RTS/CTS protocol. As expected, both maximum throughputs decrease with a_s, but the rate of decrement is slower using the RTS/CTS protocol. At sufficiently large a_s, using the RTS/CTS protocol can deliver up to five times the ALOHA throughput. Hence, the use of the RTS/CTS protocol is especially important when the source and destination are far apart.

6.6 Conclusion

We developed a mathematical framework for the behavior of the RTS/CTS protocol based on successive captures. In particular we studied the evolution of the interfering traffic densities for successive phases in the protocol cycle. This allowed us to derive new expressions for the capture probabilities of the RTS, CTS and data packets. The statistical dependence of the capture probabilities has been calculated considering a chain of capture probabilities on the desired link, probabilities of potential interferers missing the inhibition messages, and the resulting interfering traffic intensity. We have shown that for ad-hoc networks, the popular assumption that the capture probability of the RTS is representative for the probability of a successful cycle is no longer accurate when we perform rate optimization.

[4]The throughput for the QS channel for $P \to \infty$ is obtained by a semi-analytical method. For completeness, the details are given in Appendix 6.E.

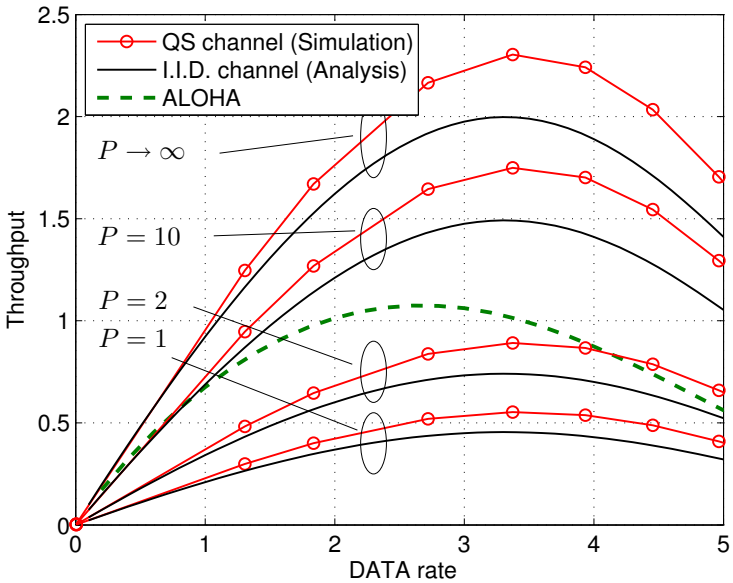


Fig. 6.7: Slot-by-slot DATA detection: throughput $\bar{s}$ vs rate of DATA slots R_{D} for various DATA slot lengths P.

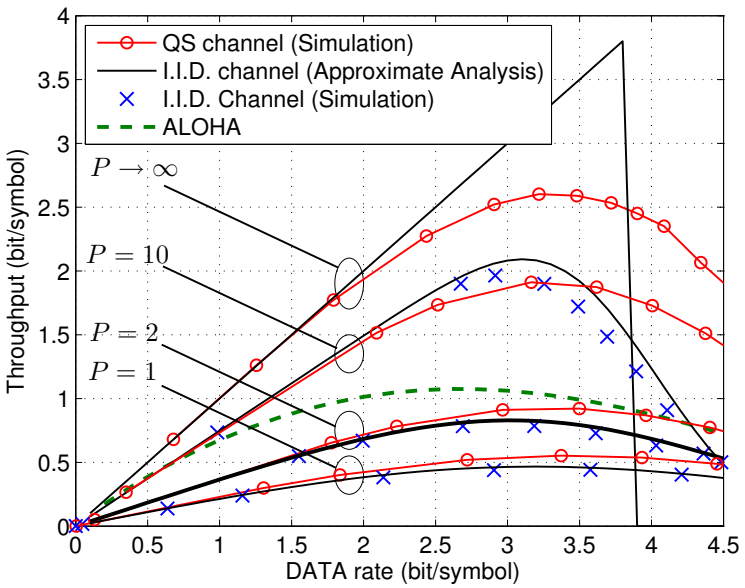


Fig. 6.8: Entire-packet DATA detection: throughput $\bar{s}$ vs rate of DATA slots R_{D} for various DATA slot lengths P.

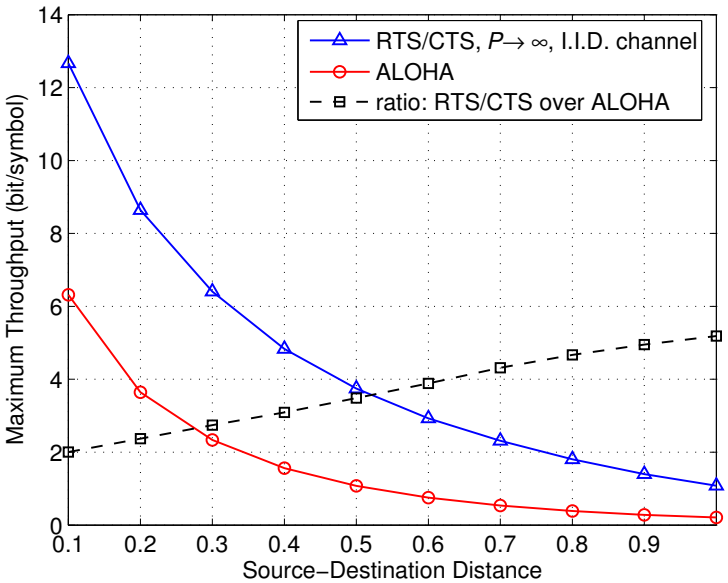


Fig. 6.9: Maximum throughput achieved at varying source-destination distance by using the RTS/CTS or ALOHA protocol.

Numerical results suggest that the rates for transmitting RTS and CTS packets are preferably around three times lower than the rates for transmitting the data payload; this significantly improves throughput. We observe that using the RTS/CTS protocol is especially worthwhile, as compared to the ALOHA protocol, when the data length is long and the source is far from the destination.

Appendix 6.A Derivation of (6.27)

We consider the RTS phase. We divide the operating region $\mathcal{A}$ into $K(\to\infty)$ non-overlapping regions, $\mathcal{A}_k$, each of area $\delta A(\to 0)$. An interferer at $\mathbf{a}_k \in \mathcal{A}_k$ transmits with probability $\Pr(k_{\rm R}\text{ on}), k = 1, \cdots, K$. By definition of the Poisson process, the traffic intensity in the RTS phase can be expressed as

$$\mathcal{G}_{\rm R}(\mathbf{a}_k) = \lim_{\delta A\to 0} \frac{\Pr(k\text{ on})}{\delta A}. \tag{6.34}$$

Knowing that the RTS is recovered by the D, the *a posteriori* traffic intensity is then

$$\begin{aligned}\mathcal{G}_{\rm R}(\mathbf{a}_k|\mathcal{C}_{\rm R}) &= \lim_{\delta A\to 0} \frac{\Pr(k\text{ on}|\mathcal{C}_{\rm R})}{\delta A} \\ &= \lim_{\delta A\to 0} \frac{\Pr(k\text{ on}|\mathcal{C}_{\rm R})}{\Pr(k\text{ on})}\, \mathcal{G}_{\rm R}(\mathbf{a}_k),\end{aligned} \tag{6.35}$$

by substituting (6.34). It is shown in [53, (29)] that

$$\frac{\Pr(k\text{ on}|\mathcal{C}_{\rm R})}{\Pr(k\text{ on})} = \frac{1 - W(z_{\rm R}, a_s, |\mathbf{a}_k - \mathbf{a}_{\mathsf{D}}|)}{1 - W(z_{\rm R}, a_s, |\mathbf{a}_k - \mathbf{a}_{\mathsf{D}}|)\Pr(k\text{ on})}, \tag{6.36}$$

where $a_s = |\mathbf{a}_{\mathsf{S}} - \mathbf{a}_{\mathsf{D}}|$. Note that $\lim_{\delta A\to 0}\Pr(k\text{ on}) = 0$; otherwise $\mathcal{G}_{\rm R}(\mathbf{a}_k|\mathcal{C}_{\rm R}) \to \infty$ in (6.34). So, (6.36) converges to $1 - W(z_{\rm R}, a_s, |\mathbf{a}_k - \mathbf{a}_{\mathsf{D}}|)$ as $\delta A \to 0$. Hence, (6.35) becomes $\mathcal{G}_{\rm R}(\mathbf{a}_k|\mathcal{C}_{\rm R}) = (1-W(z_{\rm R}, a_s, |\mathbf{a}_k-\mathbf{a}_{\mathsf{D}}|))G_o$ since $\mathcal{G}_{\rm R}(\mathbf{a}) = G_o$, hence completing the derivation.

Appendix 6.B Derivation of (6.29)

We use similar arguments as in Appendix 6.A to derive (6.29), (6.30). To show (6.29), we consider the CTS phase and focus on an interferer at $\mathbf{a}_k$ that occupies a small area δA. The event "k on" represents that interferer k transmits in the *CTS phase*. By definition, the traffic intensity given $\mathcal{C}_{\rm R}$ is

$$\mathcal{G}_{\rm C}(\mathbf{a}_k|\mathcal{C}_{\rm R}) = \lim_{\delta A\to 0} \frac{\Pr(k\text{ on}|\mathcal{C}_{\rm R})}{\delta A}. \tag{6.37}$$

In the CTS phase, the traffic intensity at $\mathbf{a}_k$ given $\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}$ is

$$\begin{aligned}\mathcal{G}_{\rm C}(\mathbf{a}_k|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}) &= \lim_{\delta A\to 0} \frac{\Pr(k\text{ on}|\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C})}{\delta A} \\ &= \lim_{\delta A\to 0} \frac{\Pr(k\text{ on}|\mathcal{C}_{\rm C}, \mathcal{C}_{\rm R})}{\Pr(k\text{ on}|\mathcal{C}_{\rm R})}\, \mathcal{G}_{\rm C}(\mathbf{a}_k|\mathcal{C}_{\rm R})\end{aligned} \tag{6.38}$$

by using (6.37). Similar to the derivations in [53, (29)], we get

$$\frac{\Pr(k\text{ on}|\mathcal{C}_{\rm C}, \mathcal{C}_{\rm R})}{\Pr(k\text{ on}|\mathcal{C}_{\rm R})} = \frac{1 - W(z_{\rm C}, a_S, |\mathbf{a}_k - \mathbf{a}_{\mathsf{S}}|)}{1 - W(z_{\rm C}, a_S, |\mathbf{a}_k - \mathbf{a}_{\mathsf{S}}|)\Pr(k\text{ on}|\mathcal{C}_{\rm R})} \tag{6.39}$$

which approaches its numerator as $\delta A \to 0$, and hence we obtain (6.29).

To obtain (6.30), we consider instead the RTS phase, i.e., the event "k on" represents that interferer k transmits in the *RTS phase*. By definition, the traffic intensity given $\mathcal{C}_{\rm R}, \mathcal{C}_{\rm C}$ is

$$\begin{aligned}\mathcal{G}_{\rm R}(\mathbf{a}_k|\mathcal{C}_{\rm R},\mathcal{C}_{\rm C}) &= \lim_{\delta A\to 0}\frac{\Pr(k \text{ on}|\mathcal{C}_{\rm R},\mathcal{C}_{\rm C})}{\delta A}\\ &= \lim_{\delta A\to 0}\frac{\Pr(k \text{ on}|\mathcal{C}_{\rm R})}{\delta A}\frac{\Pr(k \text{ on}|\mathcal{C}_{\rm C},\mathcal{C}_{\rm R})}{\Pr(k \text{ on}|\mathcal{C}_{\rm R})}\\ &= \mathcal{G}_{\rm R}(\mathbf{a}_k|\mathcal{C}_{\rm R})\,(1-W(z_{\rm C},a_S,|\mathbf{a}_k-\mathbf{a}_{\mathsf{S}}|))\end{aligned}$$

by substituting (6.35) and following the same derivation as for (6.39). Thus, we obtain (6.30).

Appendix 6.C Derivation of (6.32)

The conditional DATA capture probability is given by

$$\Pr(\mathcal{C}_{\rm D}|\mathbb{T}_{\rm cycle}) = \Pr\left(\frac{\gamma_{\rm R}^{\rm sig}}{\gamma_{\rm R}^{\rm int}}>z_{\rm R},\frac{\gamma_{\rm C}^{\rm sig}}{\gamma_{\rm C}^{\rm int}}>z_{\rm C},\frac{\gamma_{\rm D}^{\rm sig}}{\gamma_{\rm D}^{\rm int}}>z_{\rm D}\Big|\mathbb{T}_{\rm cycle}\right) \tag{6.40}$$

for any channel, by using (6.5) and (6.6). For conciseness, we denote the signal powers as $\boldsymbol{\gamma}^{\rm sig} \triangleq [\gamma_{\rm R}^{\rm sig},\gamma_{\rm C}^{\rm sig},\gamma_{\rm D}^{\rm sig}]$ and the total interference powers as $\boldsymbol{\gamma}^{\rm int} \triangleq [\gamma_{\rm R}^{\rm int},\gamma_{\rm C}^{\rm int},\gamma_{\rm D}^{\rm int}]$. Further, we denote the dummy variables used for integration as $\mathbf{y}=[y_1,y_2,y_3],\mathbf{x}=[x_1,x_2,x_3]$. By an appropriate change of variables, (6.40) can be written as

$$\Pr(\mathcal{C}_{\rm D}|\mathbb{T}_{\rm cycle}) = \int_0^\infty\int_0^\infty\int_0^\infty f_{\boldsymbol{\gamma}^{\rm int}}(\mathbf{x}|\mathbb{T}_{\rm cycle})\mathcal{K}(\mathbf{x})\,\mathrm{d}\mathbf{x}, \tag{6.41}$$

where we define

$$\mathcal{K}(\mathbf{x}) \triangleq \int_{y_1=z_{\rm R}x_1}^\infty\int_{y_2=z_{\rm C}x_3}^\infty\int_{y_3=z_{\rm D}x_3}^\infty f_{\boldsymbol{\gamma}^{\rm sig}}(\mathbf{y})\,\mathrm{d}\mathbf{y}. \tag{6.42}$$

The pdfs of $\boldsymbol{\gamma}^{\rm sig}$ for I.I.D., QSNR and QS channels are given by

$$\begin{aligned}f_{\boldsymbol{\gamma}^{\rm sig}}(\mathbf{y}) &= f_{\gamma_{\rm R}^{\rm sig}}(y_1)f_{\gamma_{\rm C}^{\rm sig}}(y_2)f_{\gamma_{\rm D}^{\rm sig}}(y_3),\\ f_{\boldsymbol{\gamma}^{\rm sig}}(\mathbf{y}) &= f_{\gamma_{\rm R}^{\rm sig}}(y_1)f_{\gamma_{\rm C}^{\rm sig}}(y_2)\delta(y_1-y_3),\\ f_{\boldsymbol{\gamma}^{\rm sig}}(\mathbf{y}) &= f_{\gamma_{\rm R}^{\rm sig}}(y_1)\delta(y_1-y_2)\delta(y_1-y_3),\end{aligned}$$

respectively. For Rayleigh-fading channels, (6.42) evaluates as

$$\begin{aligned}\mathcal{K}(\mathbf{x};\ \text{I.I.D.}) &= e^{-\eta_1/\bar{\gamma}}, \quad \eta_1 = z_{\rm R}x_1+z_{\rm C}x_2+z_{\rm D}x_3,\\ \mathcal{K}(\mathbf{x};\ \text{QSNR}) &= e^{-\eta_2/\bar{\gamma}}, \quad \eta_2 = \max\{z_{\rm R}x_1,z_{\rm D}x_3\}+z_{\rm C}x_2,\\ \mathcal{K}(\mathbf{x};\ \text{QS}) &= e^{-\eta_3/\bar{\gamma}}, \quad \eta_3 = \max\{z_{\rm R}x_1,z_{\rm C}x_2,z_{\rm D}x_3\}.\end{aligned}$$

Clearly $\eta_1 \geq \eta_2 \geq \eta_3$, hence $\mathcal{K}(\mathbf{x};\ \text{I.I.D.}) \leq \mathcal{K}(\mathbf{x};\ \text{QSNR}) \leq \mathcal{K}(\mathbf{x};\ \text{QS})$. It follows from (6.41) that the inequality (6.32) holds.

Appendix 6.D Derivation of (6.33)

For any *generic channel type* (GCT), which can be QS, QSNR or I.I.D., the DATA capture probability is given by

$$\begin{aligned} & \Pr(\mathcal{C}_{\mathrm{D}}; \mathrm{GCT}) \\ = & \int \Pr(\mathcal{C}_{\mathrm{D}}|\mathbb{T}_{\mathrm{cycle}}; \mathrm{GCT}) f_{\mathbb{T}_{\mathrm{cycle}}}(\mathbb{T}_{\mathrm{cycle}}; \mathrm{GCT})\, \mathrm{d}\mathbb{T}_{\mathrm{cycle}}. \end{aligned} \tag{6.43}$$

From (6.32) and (6.43), clearly (6.33) holds if the cycle traffic pdf does not depend on GCT, i.e., if

$$f_{\mathbb{T}_{\mathrm{cycle}}}(\mathbb{T}_{\mathrm{cycle}}; \mathrm{GCT}) = f_{\mathbb{T}_{\mathrm{cycle}}}(\mathbb{T}_{\mathrm{cycle}}) \tag{6.44}$$

regardless of GCT, for $\Pr(\mathcal{C}_{\mathrm{D}}|\mathbb{T}_{\mathrm{cycle}}; \mathrm{GCT}) > 0$. To show that (6.44) indeed holds, we express the pdf of the cycle traffic as

$$\begin{aligned} f_{\mathbb{T}_{\mathrm{cycle}}}(\mathbb{T}_{\mathrm{cycle}}; \mathrm{GCT}) = & \quad f_{\mathbb{T}_{\mathrm{R}}}(\mathbb{T}_{\mathrm{R}}; \mathrm{GCT}) f_{\mathbb{T}_{\mathrm{C}}}(\mathbb{T}_{\mathrm{C}}|\mathbb{T}_{\mathrm{R}}; \mathrm{GCT}) \\ & \times \quad f_{\mathbb{T}_{\mathrm{D}}}(\mathbb{T}_{\mathrm{D}}|\mathbb{T}_{\mathrm{R}}, \mathbb{T}_{\mathrm{C}}; \mathrm{GCT}). \end{aligned}$$

The DATA capture probability is positive only if past successive captures have occurred. Therefore, we can write $f_{\mathbb{T}_{\mathrm{C}}}(\mathbb{T}_{\mathrm{C}}|\mathbb{T}_{\mathrm{R}}; \mathrm{GCT})$ as $f_{\mathbb{T}_{\mathrm{C}}}(\mathbb{T}_{\mathrm{C}}|\mathbb{T}_{\mathrm{R}}, \mathcal{C}_{\mathrm{R}}; \mathrm{GCT})$ and $f_{\mathbb{T}_{\mathrm{D}}}(\mathbb{T}_{\mathrm{D}}|\mathbb{T}_{\mathrm{R}}, \mathbb{T}_{\mathrm{C}}; \mathrm{GCT})$ as $f_{\mathbb{T}_{\mathrm{D}}}(\mathbb{T}_{\mathrm{D}}|\mathbb{T}_{\mathrm{R}}, \mathbb{T}_{\mathrm{C}}, \mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}}; \mathrm{GCT})$. Regardless of GCT, knowing $\mathcal{C}_{\mathrm{R}}$ is sufficient to determine the pdf of $\mathbb{T}_{\mathrm{C}}$, and knowing $\mathcal{C}_{\mathrm{R}}$ and $\mathcal{C}_{\mathrm{C}}$ is sufficient to determine the pdf of $\mathbb{T}_{\mathrm{D}}$. Thus, (6.44) holds which completes the proof.

Appendix 6.E Obtaining Throughput in QS Channels for $P \to \infty$

We use a semi-analytical method to obtain the throughput $\bar{\bar{s}}(R_{\mathrm{D}})$ in QS channels for $P \to \infty$, as follows. We note that $\overline{\mathcal{I}}_{\mathrm{D}}$ *conditioned* on a received signal power $\gamma (= \gamma_{\mathrm{R}}^{\mathrm{sig}} = \gamma_{\mathrm{C}}^{\mathrm{sig}} = \gamma_{\mathrm{D}}^{\mathrm{sig}})$ approaches the constant $\widetilde{\mu}_{\mathcal{I}}(\gamma) \triangleq \mathbb{E}[\mathcal{I}_{\mathrm{D}}|\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}}, \gamma]$ for large P. The asymptotic throughput is then given by $\bar{\bar{s}}(R_{\mathrm{D}}) = R_{\mathrm{D}} \Pr(\widetilde{\mu}_{\mathcal{I}}(\gamma) > R_{\mathrm{D}}|\mathcal{C}_{\mathrm{R}}, \mathcal{C}_{\mathrm{C}})$, where γ is taken as the random variable. Hence, by obtaining samples of $\widetilde{\mu}_{\mathcal{I}}(\gamma)$ from Monte Carlo simulations, $\bar{\bar{s}}(R_{\mathrm{D}})$ can be computed numerically.

CHAPTER 7

Conclusion

"*Out of clutter, find simplicity. From discord, find harmony. In the middle of difficulty, lies opportunity.*"
Albert Einstein

In this dissertation, we employ a transmission strategy to deal with the time variations of the wireless channel. A transmission strategy consists of the channel state information (CSI), the adaptation algorithm and the transmission parameters. These key elements are illustrated in Figure 7.1. During the adaptation process, the adaptation algorithm uses some CSI as input, usually obtained as feedback from the receiver, and outputs an appropriate transmission parameter to be used by the transmitter. We have proposed that a pragmatic well-designed transmission strategy should be *agile*, as characterized by three key features: only a lean feedback is needed, the adaptation process is responsive, and the adaptation algorithm is simple to implement.

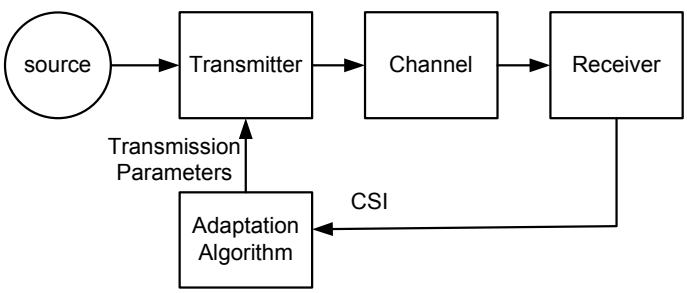


Fig. 7.1: Transmission strategy in a communication system (duplicated from Figure 1.3).

7.1 Applicability of Agile Transmission Strategies

Chapter	Channel	Physical Layer	Data Link Layer
2	multi-user	single carrier	Type I hybrid ARQ (LLC sublayer)
3	single-user	single carrier	Type II hybrid ARQ (LLC sublayer)
4, 5	single-user	multi-carrier	Type II hybrid ARQ (LLC sublayer)
6	multi-user	single carrier	RTS/CTS protocol (MAC sublayer)

Table 7.1: Channels and systems considered in this dissertation (duplicated from Table 1.1).

Chapters 2-6 are the main chapters in this dissertation. As shown in Table 7.1, we have considered single-user and multi-user channels, single-carrier and multi-carrier systems in the physical layer, and different schemes in the data link layer. Through these varied scenarios, we have demonstrated that the concept of an agile transmission strategy is widely applicable in many channels and systems.

Furthermore, we have considered agile transmission strategies across the data link layer and physical layers, and have demonstrated that the throughput can be significantly increased with lean feedback and simple adaptation algorithms. We have shown that agile transmission strategies can be implemented using cross-layer techniques, by making modifications on how the CSI is shared across the layers. Moreover, the proposal of new schemes in one layer can impact other layers. Hence, in developing agile transmission strategies we have often performed joint designs of both the physical and data link layers.

7.2 Agility of Transmission Strategy

In this dissertation, we have designed agile transmission strategies that deal with the challenge of tracking and adapting quickly to yield a significant performance gain with lean feedback and simple adaptation algorithms. To this end, our design philosophy is to start with a well-defined (non-agile) transmission strategy that requires feedback of all channel states. We then obtain an optimal or close-to-optimal solution by reducing the amount of feedback, so that that the feedback becomes lean and yet the performance is acceptable. Typically, this solution is complex to implement. Then, we proceed to to simplify the solution with little performance loss.

Consequently, the transmission strategies that we have proposed display the characteristics of agility, since a lean feedback is used, the adaptation algorithm is simple, and the adaptation process is responsive. Very often, this feedback is already available in present or future standards, and so essentially no extra feedback is required. In addition, the simplicity of the proposed adaptation algorithms encourages adoption of the transmission strategies.

The leanness, simplicity and responsiveness of the agile transmission strategies that will be discussed in Chapters 2-6 are outlined in Table 7.2.

Chapter(s)	Feedback	Adaptation Algorithm	Adaptation Process
2, 3	History of ACKs and Rates	Particle Filter / Lookup Table	Packet-by-Packet
4, 5	Ranking of Subcarrier SNRs	Oppositely-Ordered Assignment	Block-by-Block
6	RTS, CTS Signaling	Lookup Table	Cycle-by-Cycle

Table 7.2: Key features of the agile transmission strategies considered in this dissertation.

To achieve lean feedback, in Chapters 2, 3 we have used the history of past ACKs and past rates as CSI, which are in fact already available in most ARQ systems. In Chapters 4, 5, we consider OFDM systems. For OFDM systems with a large number of subcarriers, potentially many channel states need to be fed back, one for each subcarrier. Instead, we have fed back only the ranking of the SNRs of the subcarriers (or *groups* of subcarriers). Finally, in Chapter 6 we have used the standard RTS and CTS signalling as CSI and hence no additional feedback is required. In conclusion, the feedback used is lean and is often already available in current systems.

To have simple implementations of adaptation algorithms, we focus on simplifying optimal or close-to-optimal solutions. The adaptation algorithms we have proposed can mostly be implemented as a lookup table, namely in Chapters 2, 3, 6. The algorithm to determine the oppositely-ordered assignment (and its variation) in Chapter 5 is also very simple to implement, compared to an exhaustive search of the optimal assignment.

The adaptation processes we have considered are responsive to channel variations, but their degrees of responsiveness depend strongly on the system considered (single-carrier or multi-carrier) and the channel (single-user or multi-user). For each update of the CSI, the adaptation algorithm chooses the transmission parameters to pass to the transmitter. This update is performed on a packet-by-packet basis in Chapters 2, 3, on a block-by-block basis in Chapters 4, 5, and on a cycle-by-cycle basis in Chapter 6. In Chapters 4, 5, OFDM-based systems are considered and each *block* comprises a number of OFDM symbols, while in Chapter 6, the request-to-send/clear-to-send (RTS/CTS) protocol is used for channel access, and a *cycle* refers to the entire transmissions of the RTS, CTS (both for reserving the channel) and data payload.

7.3 Tradeoffs

Generally, a tradeoff has to be made among having lean CSI, a responsive adaptation process and a simple adaptation algorithm, since these characteristics are inter-related and often demand contradictory solutions. In this dissertation, usually a reasonable tradeoff is found after several rounds of explorations among different suitable choices of characteristics. For example in Chapter 2, these rounds of explorations have resulted

in a series of throughput curves, which worsen gracefully as the amount of feedback is reduced or the algorithm is simplified. Depending on the engineering applications and system resources, some of these intermediate algorithms (that incur more feedback or higher complexity) may actually be useful. Hence, there may not be just one good agile transmission strategy, but many, depending on how the tradeoff is made.

7.4 Discussions on Contributions

7.4.1 Chapter 2: Rate Adaptation using ACK Feedback

To match variations in channel conditions, in Chapter 2 we have employed rate adaptation and sought to maximize the throughput over an infinite time horizon. In order to limit the feedback, only the past ACKs and past rates are exploited. For our investigations, we have introduced a more general wireless channel model that includes multi-user interference, in which negative ACKs may be due to either a deep channel fade or excessive interference.

The complexity of this throughout maximization problem is prohibitive. As such, optimal rate schemes cannot be practically implemented, and also the maximum achievable throughput cannot be quantified. Hence, we have employed a two-pronged approach to achieve our goal. Firstly, we established a tighter upper bound, and secondly, we proposed more practical rate adaptation schemes as compared to the literature. Specifically, we have proposed the *particle-filter-based rate adaptation* (PRA), which employs the particle filter to estimate the *a posteriori* channel density, and the *rule-based rate adaptation* (RRA), which is implemented by a lookup table. Besides having better throughput performance at convergence, the PRA is useful if more responsive adaptation during initialization is desired; otherwise the RRA is more appealing with its simpler implementation. Numerical studies has shown that the PRA and RRA perform close to the newly derived upper bound over a wide range of SNRs.

7.4.2 Chapter 3: IRID ARQ Coding Scheme

To further improve the utilization of channel resources, in Chapter 3 we have employed incremental redundancy (IR) in retransmission packets. Moreover, we have proposed that incremental data (ID), i.e., additional new (information) bits not previously sent which may be encoded, is sent in the retransmission using the remaining channel resources that are not used for sending IR. This resulted in the IRID ARQ scheme. The IRID scheme is appropriate when the source always contains sufficient bits in the queue, such as in applications like multimedia streaming. To achieve an agile transmission strategy, the IRID scheme is implemented by time-multiplexing the IR and ID, and uses a lean single-bit ACK feedback (even though different types of information bits, namely the erroneous bits and the ID bits, are present). Numerical results for Rayleigh fading channels have revealed the effectiveness of the proposed

code, showing that it achieves close to the maximum possible throughput where full knowledge of the channel is available at the transmitter.

7.4.3 Chapter 4: Iterative Subcarrier Reconstruction in OFDM Systems

Since wireless channels are typically frequency-selective, in Chapter 4 we have focused on PT-OFDM systems. We sought to overcome the challenge of designing simple detectors with performance that are comparable to conventional iterative detectors. To this end, we have proposed an iterative detector that exploits differences in channel powers at different subcarriers. We have provided an analysis of the optimization of the transform coefficients and iterative method so as to maximize the minimum SNR. We analyzed the error performance if there were no error propagation and showed that the diversity is increased by one for every additional reconstruction performed. Numerical results have shown that the proposed detector has lower complexity than conventional detectors, yet has better performance. Moreover, it performs close to the optimal maximum-likelihood detector over a wide range of SNRs.

7.4.4 Chapter 5: ARQ by Subcarrier Assignment

As a further step towards designing agile transmission strategies, in Chapter 5 we have addressed the challenge of exploiting lean CSI for designing channel-aware subcarrier assignments that are simple to implement. By subcarrier assignment, we mean the assignment of ARQ subcarriers (in a retransmission) to original subcarriers (in a failed transmission). To meet this challenge, we have formulated two ARQ subcarrier assignment (ARQ-SA) problems; the second problem is slightly more general than the first. The optimality of an assignment proposed for the first problem has been proven by using the theory of majorization [42]. We have shown that the second problem is NP-hard, and thus we proposed a sub-optimum subcarrier assignment solution. Both proposed assignments are simple to implement. Furthermore, the CSI can be made leaner by performing assignment on groups of subcarriers, at negligible performance loss. Numerical results have indicated that substantial throughput improvement can be achieved by the proposed assignments, compared to fixed assignments in the literature that do not exploit CSI.

7.4.5 Chapter 6: Successive-Capture Analysis of RTS/CTS

To better understand the effects of the wireless channel on the RTS/CTS protocol, in Chapter 6 we have dropped the conventional assumption that the CTS, DATA and ACK packets are always recovered if the RTS packet is recovered. Instead, we have considered more realistic, statistically dependent *capture probabilities*, i.e., the probabilities that packets are successfully recovered, as the protocol progresses. We called this successive-capture analysis. We have observed that the RTS/CTS protocol

makes an efficient *spatial* reservation of the channel, by inhibiting only nodes that can potentially harm the reception of packets. This demonstrates that the severity of the hidden-node and exposed-node problems has been limited by the RTS/CTS protocol. Moreover, we confirmed (and quantified) that for moderately large messages, nodes can achieve a significantly higher throughput when using the RTS/CTS protocol, compared to the ALOHA protocol. From our investigations we have further obtained the optimum rate for data transmission for given RTS and CTS rates. Such rate optimization cannot be obtained without the use of successive-capture analysis, since conventional capture analysis does not adequately account for the effects of data rates on the performance of the RTS/CTS protocol.

7.5 Suggestions for Further Work

The research carried out in this thesis is by no means complete. In the course of writing this dissertation, new preliminary work on agile transmission strategies has been pursued and is briefly reported here. Suggestions for further work are also given.

7.5.1 Transmission Parameters Available for Tuning

In this dissertation, we have considered rate adaptation and subcarrier assignment, as shown in Table 7.2. Other transmission parameters that can be tuned are transmission power, the transform employed in PT-OFDM, etc. We expect that performance will improve with more parameters available for tuning. Although there are incentives to increase the number of tunable transmission parameters, more feedback may be needed for optimization and the adaptation algorithm can become more complex. Hence, a tradeoff of these issues should be made.

For OFDM systems, power can be allocated across subcarriers such that the mean power is fixed. Power allocation across subcarriers in this way can be adapted according to channel conditions. As part of an ongoing study, we investigate power allocation and subcarrier assignment for two-way relaying in [116]. The complexity of these algorithms is high and part of the future work will be to reduce this complexity.

7.5.2 Getting CSI from Environment

In this dissertation, we have considered that the CSI can be made available as feedback from the transmitter. This has been illustrated in Figure 7.1. Additionally, the CSI could be obtained as observations from the environment, as shown in Figure 7.2. This part of the CSI does not require explicit feedback from the receiver. Although the amount of feedback required remains the same, more CSI becomes available as input to the adaptation algorithm, hence improving system performance. Preliminary work is presently being conducted in this direction [62].

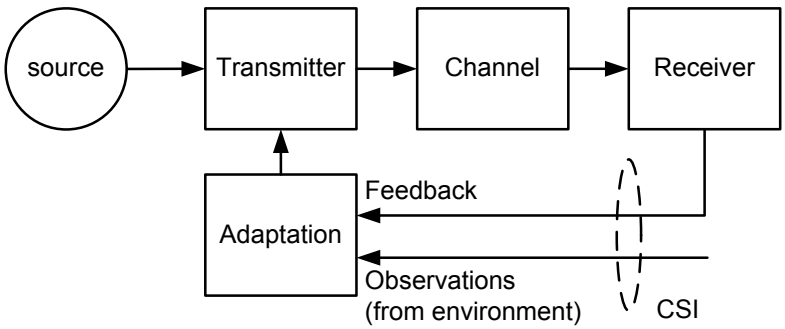


Fig. 7.2: Transmission strategy in a communication system, similar to Figure 7.1. However, the CSI here can be additionally obtained by observing the environment.

7.5.3 Different Performance Measures

In the dissertation, we have considered throughput as our main performance measure. We do not account for the stability of the wireless network, and as a result packet delay can be infinite. We also ignore the fairness issue, as such certain users may be treated more favorably. These issues are important and can be treated by appropriate assignments of cost functions (to discourage large packet delay) and system-wide utility functions (to encourage fairness). It would be insightful to evaluate the impacts that these cost functions and utility functions have on the agility of the transmission strategy. Future research could focus on these issues.

7.5.4 Subcarrier Assignments

The assignment of ARQ subcarriers in a retransmission to original subcarriers in a failed transmission has been considered in Chapter 5. If at most one ARQ subcarrier is allowed to be assigned to each original subcarrier, we have shown that the optimal subcarrier assignment is given by the *oppositely*-ordered assignment (in which the mth *strongest* ARQ subcarrier is assigned to the mth *weakest* original subcarrier). By employing the theory of majorization [42], this result holds as long as the utility function that we wish to maximize is Schur concave. Interestingly, it can be shown that many typical utility functions are indeed Schur concave. More generally, the optimal assignment obtained from the theory of majorization enjoys a certain *universal optimality*. By universal optimality, we mean that the optimal assignment remains the same even if the utility function changes within a large class of functions which includes practically meaningful functions, such as those considered in Chapter 5. Besides being useful from an implementation point of view, this property suggests that the optimal assignment is robust to different definitions of utility or performance measure considered by different areas of communication specialists.

Subcarrier assignments can also be extended to other communication systems. In relay-assisted communications, a relay helps a source to forward a packet to a des-

tination. Subcarrier assignment can be carried out if all the nodes employ OFDM. Specifically, the relay, after receiving the OFDM symbol from the source, can permute the data carried by the subcarriers before forwarding to the destination. We have considered this approach and reported the results in [94]. Interestingly, we found that the optimal assignment becomes completely reversed: the *similarly*-ordered assignment (in which the nth *strongest* relay-destination subcarrier is assigned to the nth *strongest* source-relay subcarrier) is now optimal, instead of the oppositely-ordered assignment.

In [94], the subcarrier assignment is optimized so as to maximize the sum capacity. However, it is also important to investigate the practical case of minimizing the BER. Moreover, in [94] the destination does not make use of the transmission from the source for joint decoding. Instead, the destination should also make use of the transmission from the source for joint detection to improve the detection probability. Preliminary findings for both of these issues have been reported recently in [117].

In [94,117], we consider a one-way relay, where information flows only unidirectionally, from the source to the relay and finally to the destination. Two-way relaying, where addition information flows in a reverse direction, from the destination to the relay and finally to the source, has been recently considered in the literature [118]. Applications of subcarrier assignments to the two-way relay when OFDM is used in all nodes can achieve further improvement in performance. Preliminary findings have been reported recently in [116].

In subcarrier-assignment schemes, the receiver needs to feed back the chosen assignment. This assignment is computed by the receiver based on instantaneous subcarrier SNRs. Alternatively, it is also possible to feed back only a long-term CSI, for instance average subcarrier SNRs, that is valid for a longer time scale. Then, the transmitter would compute the assignment based on this (infrequent) long-term CSI. Other practical forms of CSI should also be investigated to explore the feasibility of implementing subcarrier assignment. In particular, estimates of the instantaneous subcarrier SNRs are typically available, instead of the exact subcarrier SNRs. A natural question is whether we can use these estimates for optimizing the subcarrier assignment, in place of the exact SNRs. The framework of stochastic majorization [119] is useful for investigating such issues. Stochastic majorization is more general than majorization and is particularly appropriate for dealing with partial CSI. By applying the theory of stochastic majorization to the scenario when the transmitter knows the average SNR (rather than the instantaneous SNR), we can obtain the optimal subcarrier assignment; this new work is reported recently in [120].

7.6 Key Future Challenges and Opportunities

As technology matures and applications evolve in the future, different tradeoffs have to be made in designing agile strategies that are responsive in adaptation, yet lean in feedback and simple to implement. We give a perspective on how the future

might impact on the design of agile strategies and suggest some general research opportunities.

To provide pervasive and yet personal communication, the number of personal communicating devices will likely increase in number but reduce in size. With the increase in number of communicating nodes, the amount and variance of aggregated interference will increase and dominate the performance of communication networks. To maintain responsiveness in adaptation, more focus has to be placed on accounting for the interference level, in addition to the channel fades. Moreover, for scalability, a distributed adaptation algorithm will be more desirable, compared to a centralized one. As such, future research should focus on interference prediction and characterization, and on the distributed adaptation algorithms that uses this additional knowledge. This research can be explored in the context of cross-layer adaptation, since issues of channel fading and network interference are typically considered in the physical and data link layers.

As the number of communicating devices increases, the total amount of feedback required for adaptation increases proportionally. Hence, the use of lean feedback will become even more important in the future. To improve understanding on the impact, the optimality and the stability of the use of lean feedback, rich results found in control theory can be applied. For example, optimal adaptation policies can be obtained by dynamic programming in the field of stochastic control. However, the application of such theoretical results may have to be considered specifically for our engineering scenario. For example, the optimal policies found by dynamic programming may not be computable, and so approximations or bounds have to devised accordingly. Hence, a discriminating application of control theory that are useful for designing agile strategies is needed, but this approach provides interesting research opportunities and can lead to potentially impactful results.

Moore's law promises that complex algorithms can eventually be manufactured at a low cost. However, technological advances in battery efficiency have so far lagged behind. A highly complex algorithm operating on a battery-powered device consumes much energy, and thus experiences a short active cycle before the next round of battery recharge or battery replacement. This may make complex algorithms impractical for use in lightweight mobile devices. Hence, low energy consumption, rather than low implementation complexity, may be a more relevant measure of "simplicity" for agile strategies in the future. To illustrate the impact that it may have on the design philosophy, let us compare two devices. The first device is always listening to ascertain when is a good opportunity to send data. The second device uses a more complex algorithm to determine when is the best opportunity to listen periodically and when to transmit, but sleeps otherwise. Even though the first device consumes less peak power, its energy consumption on average may still be higher than the second device. This illustrates that a low-energy algorithm can be designed quite differently from a low-complexity algorithm. Hence, opportunities lie in pursuing agile strategies that account for the energy consumption in an explicit way.

Bibliography

[1] R. H. Katz, "Adaptation and mobility in wireless information systems," *IEEE Personal Commun.*, vol. 1, no. 1, pp. 6–17, First Quarter 1994.

[2] Q. Bi, G. L. Zysman, and H. Menkes, "Wireless mobile communications at the start of the 21st century," *IEEE Commun. Mag.*, vol. 39, no. 1, pp. 110–116, Jan. 2001.

[3] S. Ohmori, Y. Yamao, and N. Nakajima, "The future generations of mobile communications based on broadband access technologies," *IEEE Commun. Mag.*, vol. 38, no. 12, pp. 134–142, Dec. 2000.

[4] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[5] H. Zimmermann, "OSI reference model – the ISO model of architecture for Open Systems Interconnection," *IEEE Trans. Commun.*, vol. 28, no. 4, pp. 425–432, Apr. 1980.

[6] J. M. Wozencraft and M. Horstein, "Coding for two-way channels," Res. Lab. Electron., MIT, Cambridge, MA, Tech. Rep. 383, Jan. 1961.

[7] P. Sindhu, "Retransmission error control with memory," *IEEE Trans. Commun.*, vol. 25, no. 5, pp. 473–479, May 1977.

[8] J. D. J. Costello, J. Hagenauer, H. Imai, and S. B. Wicker, "Applications of error-control coding," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2531–2560, Oct. 1998.

[9] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High Speed Physical Layer in the 5 GHz Band*, IEEE 802.11a Standard, 1999.

[10] T. Rappaport, A. Annamalai, R. Buehrer, and W. Tranter, "Wireless communications: past events and a future perspective," *IEEE Commun. Mag.*, vol. 40, no. 5, pp. 148–161, May 2002.

[11] S. Shakkottai and T. R. P. Karlsson, "Cross-layer design for wireless networks," *IEEE Commun. Mag.*, vol. 41, no. 10, pp. 74–80, Oct. 2003.

[12] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Foundations and Trends in Networking*, vol. 1, no. 1, pp. 1–144, 2006.

[13] V. Srivastava and M. Motani, "Cross-layer design: a survey and the road ahead," *IEEE Commun. Mag.*, vol. 43, no. 12, pp. 112–119, Dec. 2005.

[14] V. Kawadia and P. Kumar, "A cautionary perspective on cross-layer design," *IEEE Wireless Commun.*, vol. 12, no. 1, pp. 3–11, Feb. 2005.

[15] (2004) Status of project IEEE 802.11n. [Online]. Available: http://grouper.ieee.org/groups/802/11/Reports/tgn_update.htm

[16] G. Holland, N. Vaidya, and P. Bahl, "A rate-adaptive MAC protocol for multi-hop wireless networks," in *Proc. ACM MOBICOM'01*, Jul. 2001, pp. 236–251.

[17] D. L. Goeckel, "Adaptive coding for time-varying channels using outdated fading estimates," *IEEE Trans. Commun.*, vol. 47, no. 6, pp. 844–855, Jun. 1999.

[18] Q. Liu, S. Zhou, and G. Giannakis, "Cross-layer combining of adaptive modulation and coding with truncated ARQ over wireless links," *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, pp. 1746–1755, Sep. 2004.

[19] J. del Prado Pavon and S. Choi, "Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement," in *Proc. IEEE Int. Conf. on Communications (ICC)*, vol. 2, Anchorage, AK, May 2003, pp. 1108–1113.

[20] D. Qiao, S. Choi, and K. Shin, "Goodput analysis and link adaptation for IEEE 802.11a wireless LANs," *IEEE Trans. on Mobile Computing*, vol. 1, no. 4, pp. 278–292, Oct.-Dec. 2002.

[21] A. Kamerman and L. Monteban, "WaveLAN-II: A high-performance wireless LAN for the unlicensed band," *Bell Labs Technical Journal*, pp. 118–133, Summer 1997.

[22] P. Chevillat, J. Jelitto, A. N. Barreto, and H. L. Truong, "A dynamic link adaptation algorithm for IEEE 802.11a wireless LANs," in *Proc. IEEE Int. Conf. on Communications (ICC)*, Anchorage, May 2003, pp. 1141–1145.

[23] D. Qiao and S. Choi, "Fast-responsive link adaptation for IEEE 802.11 WLANs," in *Proc. IEEE Int. Conf. on Communications (ICC)*, Seoul, Korea, May 2005, pp. 3583–3588.

[24] M. Rice and S. B. Wicker, "Adaptive error control for slowly varying channels," *IEEE Trans. Commun.*, vol. 42, no. 234, pp. 917–926, Feb/Mar/Apr 1994.

[25] A. K. Karmokar, D. V. Djonin, and V. K. Bhargava, "POMDP-based coding rate adaptation for type-I hybrid ARQ systems over fading channels with memory," *IEEE Trans. Wireless Commun.*, vol. 5, no. 12, pp. 3512–3523, Dec. 2006.

[26] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of Markov decision processes," *Mathematics of Operations Research*, vol. 12, no. 3, pp. 441–450, 1987.

[27] N. Gordon, D. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, no. 2, pp. 107–113, Apr. 1993.

[28] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, vol. 36, no. 4, pp. 389–400, Apr. 1998.

[29] D. N. Rowitch and L. B. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatible punctured turbo (RCPT) codes," *IEEE Trans. Commun.*, vol. 48, no. 6, pp. 948–959, Jun. 2000.

[30] S. Sesia, G. Caire, and G. Vivier, "Incremental redundancy hybrid ARQ schemes based on low-density parity-check codes," *IEEE Trans. Commun.*, vol. 52, no. 8, pp. 1311–1321, Aug. 2004.

[31] J. Bingham, "Multicarrier modulation for data transmission: an idea whose time has come," *IEEE Commun. Mag.*, vol. 28, no. 5, pp. 5–14, May 1990.

[32] A. Ghosh, D. R. Wolter, J. G. Andrews, and R. Chen, "Broadband wireless access with WiMax/802.16: current performance benchmarks and future potential," *IEEE Commun. Mag.*, vol. 43, no. 2, pp. 129–136, Feb. 2005.

[33] *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation*, 3GPP Std. 3GPP TS 36.211, Rev. 8.0.0, Sep. 2007. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/36211.htm

[34] Y.-P. Lin and S.-M. Phoong, "BER minimized OFDM systems with channel independent precoders," *IEEE Trans. Signal Process.*, vol. 51, no. 9, pp. 2369–2380, Sep. 2003.

[35] Z. Liu, Y. Xin, and G. B. Giannakis, "Space-time-frequency coded OFDM over frequency-selective fading channels," *IEEE Trans. Signal Process.*, vol. 50, no. 10, pp. 2465–2476, Oct. 2002.

[36] Y. Xin, Z. Wang, and G. B. Giannakis, "Space-time diversity systems based on linear constellation precoding," *IEEE Trans. Wireless Commun.*, vol. 2, no. 2, pp. 294–309, Mar. 2003.

[37] M. K. Varanasi and B. Aazhang, "Multistage detection in asynchronous code-division multiple-access communications," *IEEE Trans. Commun.*, vol. 38, no. 4, pp. 509–519, Apr. 1990.

[38] ——, "Near-optimum detection in synchronous code-division multiple-access systems," *IEEE Trans. Commun.*, vol. 39, no. 5, pp. 725–736, May 1991.

[39] T. Kumagai, M. Mizoguchi, T. Onizawa, H. Takanashi, and M. Morikura, "A maximal ratio combining frequency diversity ARQ scheme for OFDM signals," in *Proc. IEEE Int. Conf. Personal, Indoor and Mobile Radio Communications*, vol. 2, Sep. 1998, pp. 528–532.

[40] M. Gidlund and P. Ahag, "Enhanced HARQ scheme based on rearrangement of signal constellations and frequency diversity for OFDM systems," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, vol. 1, May 2004, pp. 500–504.

[41] X. Peng, F. Chin, and A. S. Madhukumar, "Performance studies of a VSF-OFCDM system using a symbol relocated scheme during retransmission," in *Proc. IEEE Wireless Communications and Networking Conference*, vol. 2, Mar. 2005, pp. 1138–1143.

[42] A. W. Marshall and I. Olkin, *Inequalities: Theory of Majorization and Its Applications*. Academic Press Inc., 1979.

[43] N. Abramson, "The ALOHA system – Another alternative for computer communications," in *Proc. Fall Joint Computer Conf., AFIPS Conf.*, vol. 37, 1970.

[44] L. Kleinrock and F. Tobagi, "Packet switching in radio channels: Part I – Carrier sense multiple-access modes and their throughput-delay characteristics," *IEEE Trans. Commun.*, vol. 23, no. 12, pp. 1400–1416, Dec. 1975.

[45] C. van der Plas and J. P. M. G. Linnartz, "Stability of mobile slotted aloha network with Rayleigh fading, shadowing, and near-far effect," *IEEE Trans. Veh. Technol.*, vol. 39, no. 4, pp. 359–366, Nov. 1990.

[46] P. Karn, "MACA – a new channel access method for packet radio," in *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, Sep. 1990, pp. 134–140.

[47] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A media access protocol for wireless LAN's," in *Proc. SIGCOMM'94 Conf. on Communications Architectures, Protocols and Applications*, Aug. 1994, pp. 212–225.

[48] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE 802.11 Standard, 1999.

[49] S. Ray, J. B. Carruthers, and D. Starobinski, "RTS/CTS-induced congestion in ad hoc wireless LANs," in *IEEE WCNC 2003*, New Orleans, LA, Mar. 2003, pp. 1516–1521.

[50] K. Xu, M. Gerla, and S. Bae, "Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 1, pp. 107–123, Jul. 2003.

[51] G. Bianchi, "IEEE 802.11 – Saturation throughput analysis," *IEEE Commun. Lett.*, vol. 2, no. 12, pp. 318–320, Dec. 1998.

[52] J. P. M. G. Linnartz, R. Hekmat, and R.-J. Venema, "Near-far effects in land mobile random access networks with narrow-band Rayleigh fading channels," *IEEE Trans. Veh. Technol.*, vol. 41, no. 1, pp. 77–90, Feb. 1992.

[53] J. P. M. G. Linnartz, "Slotted ALOHA land-mobile radio networks with site diversity," *IEE Proc.-I*, vol. 139, no. 1, pp. 58–70, Feb. 1992.

[54] Z. Hadzi-Velkov and B. Spasenovski, "Capture effect in IEEE 802.11 basic service area under influence of Rayleigh fading and near/far effect," in *Proc. 13th IEEE Personal, Indoor and Mobile Radio Communications*, vol. 1, Lisbon, Portugal, Sep. 2002, pp. 172–176.

[55] J. H. Kim and J. K. Lee, "Capture effects of wireless CSMA/CA protocols in Rayleigh and shadow fading channels," *IEEE Trans. Veh. Technol.*, vol. 48, no. 4, pp. 1277–1286, Jul. 1999.

[56] S. H. Y. Wong, S. Lu, H. Yang, and V. Bharghavan, "Robust rate adaptation for 802.11 wireless networks," in *Proc. ACM MOBICOM'06*, 2006, pp. 146–157.

[57] J. Kim, S. Kim, S. Choi, and D. Qiao, "CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs," in *Proc. INFOCOM'06*, April 2006, pp. 1–11.

[58] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, "Opportunistic media access for multirate ad hoc networks," in *Proc. ACM MOBICOM'02*, 2002, pp. 24–35.

[59] H. Holma and A. Toskala, Eds., *HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications.* West Sussex, England: Wiley, 2006.

[60] C. C. Tan and N. C. Beaulieu, "On first-order Markov modeling for the Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 48, no. 12, pp. 2032–2040, Dec. 2000.

[61] ——, "Infinite series representations of the bivariate Rayleigh and Nakagamim distributions," *IEEE Trans. Commun.*, vol. 45, no. 10, pp. 1159–1161, Oct. 1997.

[62] T. van Berkel, J. P. Linnartz, and C. K. Ho, "Compensated estimators for characterizing interference in a Rayleigh fading environment," in *Proc. Fourteenth IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*, Delft, The Netherlands, Nov. 2007.

[63] T. P. Minka, "From hidden Markov models to linear dynamical systems," Tech. Rep., revised 7/18/99.

[64] J. Pearl, *Causality: Models, Reasoning, and Inference.* Cambridge, U.K.: Cambridge University Press, 2000.

[65] D. P. Bertsekas, *Dynamic Programming and Optimal Control Vol. 1.* Belmont, MA: Athena Scientific, 1995.

[66] M. Littman, A. Cassandra, and L. Kaelbling, "Learning policies for partially observable environments: scaling up," in *Proc. International Conference on Machine Learning*, 1995, pp. 362–370.

[67] M. I. Jordan, *An Introduction to Probabilistic Graphical Models.* In preparation, 2003.

[68] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[69] L. Lin, R. D. Yates, and P. Spasojevic, "Adaptive transmission with discrete code rates and power levels," *IEEE Trans. Commun.*, vol. 51, no. 12, pp. 2115–2125, Dec. 2003.

[70] G. Caire and D. Tuninetti, "The throughput of hybrid-ARQ protocols for the Gaussian collision channel," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1971–1988, Jul. 2001.

[71] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* John Wiley & Sons, Inc., 1991.

[72] R. van Nee and R. Prasad, *OFDM Wireless Multimedia Communications.* London: Artech House, 2000.

[73] N. Yee, J. P. M. G. Linnartz, and G. Fettweis, "Multi-carrier-CDMA in indoor wireless networks," in *Proc. IEEE Int. Conf. Personal, Indoor and Mobile Radio Communications*, Yokohama, Japan, Sep. 1993, pp. 109–113.

[74] R. van Nee, G. Awater, M. Morikura, H. Takanashi, M. Webster, and K. Halford, "New high-rate wireless LAN standards," *IEEE Commun. Mag.*, vol. 37, no. 12, pp. 82–88, Dec. 1999.

[75] J. Proakis, *Digital Communications*, 3rd ed. Singapore: McGraw-Hill, 1995.

[76] K. Fazel, "Performance of CDMA/OFDM for mobile communication systems," in *Proc. IEEE Second Int. Conf. Universal Personal Commun.*, vol. 2, Ottawa, Ont., Canada, Oct. 1993, pp. 975–979.

[77] V. Nangia and K. Baum, "Experimental broadband OFDM system: field results for OFDM and OFDM with frequency domain spreading," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, vol. 1, Vancouver, Canada, Sep. 2002, pp. 223–227.

[78] Z. Dlugaszewski and K. Wesolowski, "WHT/OFDM - an improved OFDM transmission method for selective fading channels," in *Symposium on Communications and Vehicular Technology*, Leuven, Belgium, Oct. 2000, pp. 144–149.

[79] Z. Lei, Y. Wu, C. K. Ho, S. Sun, P. He, and Y. Li, "Iterative detection for Walsh-Hadamard transformed OFDM," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, vol. 1, Jeju, Korea, Apr. 2003, pp. 637–640.

[80] H. Sari, G. Karam, and I. Jeanclaud, "Frequency-domain equalization of mobile radio and terrestrial broadcast channels," in *Proc. IEEE GLOBECOM*, vol. 1, San Francisco, CA, 1994, pp. 1–5.

[81] D. Falconer, S. L. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson, "Frequency domain equalization for single-carrier broadband wireless system," *IEEE Commun. Mag.*, vol. 40, no. 4, pp. 58–66, Apr. 2002.

[82] A. Scaglione, G. B. Giannakis, and S. Barbarossa, "Redundant filterbank precoders and equalizers Part I: Unification and optimal designs," *IEEE Trans. Signal Process.*, vol. 47, no. 7, pp. 1988–2006, Jul. 1999.

[83] Y. Ding, T. N. Davidson, Z.-Q. Luo, and K. M. Wong, "Minimum BER block precoders for zero-forcing equalization," *IEEE Trans. Signal Process.*, vol. 51, no. 9, pp. 2410–2423, Sep. 2003.

[84] A. Bury, J. Egle, and J. Lindner, "Diversity comparison of spreading transforms for multicarrier spread spectrum transmission," *IEEE Trans. Commun.*, vol. 51, no. 5, pp. 774–781, May 2003.

[85] D. A. Wiegandt, Z. Wu, and C. R. Nassar, "High-throughput, high-performance OFDM via pseudo-orthogonal carrier interferometry spreading codes," *IEEE Trans. Commun.*, vol. 51, no. 7, pp. 1123–1134, Jul. 2003.

[86] M. Debbah, W. Hachem, P. Loubaton, and M. de Courville, "MMSE analysis of certain large isometric precoded systems," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1293–1311, May 2003.

[87] M. Park, H. Jun, J. Cho, N. Cho, D. Hong, and C. Kang, "PAPR reduction in OFDM transmission using Hadamard transform," in *Proc. IEEE Int. Conf. on Communications (ICC)*, vol. 1, New Orleans, LA, Jun. 2000, pp. 430–433.

[88] Y. Wu, C. K. Ho, and S. Sun, "On some properties of Walsh-Hadamard transformed OFDM," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, vol. 4, Vancouver, BC, Sep. 2002, pp. 2096–2100.

[89] S. Hara and R. Prasad, "Overview of multicarrier CDMA," *IEEE Commun. Mag.*, vol. 35, no. 12, pp. 126–133, Dec. 1997.

[90] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1456–1467, Jul. 1999.

[91] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels: A Unified Approach to Performance Analysis*, 1st ed. New York: Wiley, 2000.

[92] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.

[93] A. Hottinen and T. Heikkinen, "Optimal subchannel assignment in a two-hop OFDM relay," in *Proc. IEEE 8th Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Helsinki, Finland, Jun. 2007, pp. 1–5.

[94] A. Pandharipande and C. K. Ho, "Spectrum pool reassignment for a cognitive OFDM-based relay system," in *Proc. 2nd International Conf. on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, Orlando, FL, Aug. 2007, pp. 90–94.

[95] D. P. Palomar and Y. Jiang, "MIMO transceiver design via majorization theory," *Foundations and Trends in Communications and Information Theory*, vol. 3, no. 4-5, pp. 331–551, 2006.

[96] E. Jorswieck and H. Boche, "Majorization and matrix-monotone functions in wireless communications," *Foundations and Trends in Communications and Information Theory*, vol. 3, no. 6, pp. 553–701, 2007.

[97] D. Palomar, J. Cioffi, and M. Lagunas, "Joint Tx-Rx beamforming design for multicarrier MIMO channels: a unified framework for convex optimization," *IEEE Trans. Signal Process.*, vol. 51, no. 9, pp. 2381–2401, Sep. 2003.

[98] A. Pandharipande and S. Dasgupta, "Optimum DMT-based transceivers for multiuser communications," *IEEE Trans. Commun.*, vol. 51, no. 12, pp. 2038–2046, Dec. 2003.

[99] L. Cai, Y. Wan, P. Song, and L. Gui, "Improved HARQ scheme using channel quality feedback for OFDM systems," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, vol. 4, May 2004, pp. 17–19.

[100] C. K. Ho, Z. Lei, S. Sun, and W. Yan, "Iterative detection for pretransformed OFDM by subcarrier reconstruction," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2842–2854, Aug. 2005.

[101] D. Knuth, *The Art of Computer Programming*, 2nd ed. Addison-Wesley, 1998.

[102] B. L. Deuermeyer, D. K. Friesen, and M. A. Langston, "Scheduling to maximize a multiprocessor system," *SIAM J. Alg. Discrete Meth.*, vol. 3, no. 2, pp. 190–196, Jun. 1982.

[103] M. Abramowitz and I. A. Stegun, Eds., *Handbook of Mathematical Functions*. New York: Dover, 1972.

[104] R. G. Gallager, *Discrete Stochastic Processes*. Boston: Kluwer Academic Publishers, 1996.

[105] (2006) GLPK (GNU linear programming kit). [Online]. Available: http://www.gnu.org/software/glpk

[106] *IEEE J. Sel. Areas Commun., Special Issue on Wireless Ad Hoc Networks*, vol. 17, Aug. 1999.

[107] N. Abramson, "The throughput of packet broadcasting channels," *IEEE Trans. Commun.*, vol. 25, no. 1, pp. 117–128, Jan. 1977.

[108] J. Arnbak and W. van Blitterswijk, "Capacity of slotted ALOHA in Rayleigh-fading channels," *IEEE J. Sel. Areas Commun.*, vol. 5, no. 2, pp. 261–269, Feb. 1987.

[109] E. S. Sousa, "Performance of a spread spectrum packet radio network link in a Poisson field of interferers," *IEEE Trans. Inf. Theory*, vol. 38, no. 6, pp. 1743–1754, Nov. 1992.

[110] R. Knopp and P. A. Humblet, "On coding for block fading channels," *IEEE Trans. Inf. Theory*, vol. 46, no. 1, pp. 189–205, Jan. 2000.

[111] L. H. Ozarow, S. Shamai, and A. D. Wyner, "Information theoretic considerations for cellular mobile radio," *IEEE Trans. Veh. Technol.*, vol. 43, no. 2, pp. 359–378, May 1994.

[112] H. Yin and S. Alamouti, "OFDMA: A broadband wireless access technology," in *Proc. IEEE Sarnoff Symposium*, Mar. 2006.

[113] *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements Enhancements*, IEEE 802.11e Standard, Nov. 2005.

[114] P. A. W. Lewis and G. S. Shedler, "Simulation of nonhomogeneous Poisson processes by thinning," *Naval Res. Logistics Quart*, vol. 26, pp. 403–413, 1979.

[115] C. K. Ho and J. P. M. G. Linnartz, "Analysis of the RTS/CTS multiple access scheme with capture effect," in *Proc. 17th IEEE Personal, Indoor and Mobile Radio Communications*, Helsinki, Finland, Sep. 2006.

[116] C. K. Ho, R. Zhang, and Y. Liang, "Two-way relaying over OFDM: Optimized tone permutation and power allocation," in *Proc. IEEE Int. Conf. on Communications (ICC)*, Beijing, China, May 2008.

[117] C. K. Ho and A. Pandharipande, "BER minimization in relay-assisted OFDM systems by subcarrier permutation," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, Singapore, Apr. 2008.

[118] S. J. Kim, P. Mitran, and V. Tarokh, "Performance bounds for bidirectional coded cooperation protocols," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 5235–5241, Nov. 2008.

[119] C.-S. Chang, J. G. Shanthikumar, and D. D. Yao, "Stochastic convexity and stochastic majorization," in *Stochastic Modeling and Analysis of Manufacturing Systems*, D. D. Yao, Ed. New York: Springer-Verlag, 1994, ch. 5, pp. 188–231.

[120] A. Pandharipande and C. K. Ho, "Stochastic spectrum pool reassignment for cognitive relay systems," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*, Las Vegas, Nevada, USA, Apr. 2008, pp. 588–592.

Samenvatting

In de voorbije jaren is veel onderzoek gedaan aan draadloze communicatie. Het gaat bij deze vorm van communicatie om het versturen van informatie vanaf een zender over een draadloos kanaal naar een ontvanger. Omdat draadloze communicatiesystemen geen bekabeling vereisen, kunnen deze eenvoudig geïnstalleerd worden. Mobiele communicatie, d.w.z. draadloze communicatie met draagbare apparatuur, gaat nog een stap verder vanwege de mogelijkheid tot communicatie onderweg.

Het geweldige succes van draadloze communicatie heeft geleid tot een enorme vraag naar spectrum, waardoor dit schaars is geworden. Tegelijkertijd worden breedbandige toepassingen, die de overdracht van grote hoeveelheden data vereisen, bijvoorbeeld video streaming, steeds populairder. De toename van het aantal gebruikers resulteert ook in een intensief spatiëel hergebruik, d.w.z. een groot aantal gelijktijdige transmissies binnen een beperkte oppervlakte. Om een hoge spectrale efficiëntie te bereiken is het belangrijk om het gebruik van de *radio resources* in tijd, frequentie en ruimte te optimaliseren. Dit is geen eenvoudige opgave door het dynamische gedrag van het draadloze kanaal; zelfs kleine veranderingen in de omgeving kunnen al een grote verandering van amplitude en fase van de ontvangen signalen teweeg brengen. Deze veranderingen van de signalen zijn in mobiele communicatie zeer prominent, omdat ook de zender en ontvanger dan normaal gesproken bewegen. In het geval van multi-user communicatie varieert de kwaliteit van de ontvangen signalen nog sterker, door de aanwezigheid van een variabel aantal interfererende gebruikers op hetzelfde draadloze kanaal.

De zender is doorgaans niet voortdurend op de hoogte van de huidige toestand van het kanaal. Maar een beperkte vorm van *kanaalinformatie* (*channel state information* of CSI) is vaak wel voorhanden, op basis van expliciete feedback van de ontvanger, of door te "luisteren" naar de omgeving. Op basis van deze CSI kunnen de transmissieparameters (zoals datasnelheid, zendvermogen, frequentie en tijdstip van verzenden) zo

gekozen worden dat de netwerkprestaties (bijv. gemeten als de gemiddelde datasnelheid, zendvermogen over een langere periode) worden geoptimaliseerd. Het aanpassen van de transmissieparameters vereist een goed ontworpen *adaptatie-algoritme*. De combinatie van de gebruikte CSI, de verzameling van toegestane transmissieparameters en het adaptatie-algoritme vormen samen de *transmissie strategie*. Voor een optimaal gebruik van de radio resources is het belangrijk om deze strategie goed te ontwerpen.

In draadloze communicatie wordt het steeds gebruikelijker om niet alle details van het systeem in een standaard vast te leggen. In plaats daarvan wordt slechts de basis van het systeem gespecificeerd, met de mogelijkheid om niet-standaard toevoegingen en verbeteringen toe te passen. Hierdoor is het mogelijk de transmissieparameters statisch of dynamisch te optimaliseren. Een voorbeeld hiervan is de nieuwe 802.11n standaard. Bij implementatie van een 802.11n systeem kan de fabrikant zelf bepalen hoe de modulatie-constellatie, de code rate, pakket-aggregatie, strategie van automatic repeat requests (ARQ) en MIMO transmissiemodus wordt gekozen. Bovendien kunnen de producten in de loop van de tijd verbeteren doordat adaptatie-algoritmes verbeterd worden ten opzichte van eerdere implementaties. Deze vrijheid levert tevens de mogelijkheid voor een fabrikant om zich te onderscheiden van de concurrentie. In dit proefschrift zoeken we naar mogelijkheden om deze vrijheidsgraden te benutten als basis voor aantrekkelijke transmissiestrategieën. De inzichten die we zo verwerven kunnen ook van pas komen bij het ontwerp van toekomstige cognitieve draadloze systemen, waarbij communicatieapparatuur dynamisch en adaptief beslist over hoe het draadloze medium te gebruiken.

In draadloze communicatie, en in het bijzonder in mobiele communicatie, wordt de aantrekkelijkheid van een transmissiestrategie met name bepaald door zijn *agility*. Onder een *agile strategy* verstaan we een "behendige" strategie die tegelijkertijd *weinig feedback* nodig heeft, en *responsief* en *eenvoudig* is:

1. Weinig feedback: het is essentieel dat de benodigde informatieoverdracht van de ontvanger naar de zender klein is: De overhead zou verwaarloosbaar moeten zijn ten opzichte van de verbetering in de hoeveelheid verzonden data ten gevolge van deze feedback
2. Responsieve adaptatie: het systeem moet in staat zijn om snel en adequaat reageren op een veranderende draadloze omgeving
3. Eenvoudige implementatie: De complexiteit van het adaptatie-algoritme moet laag genoeg zijn voor implementatie in een mobiel apparaat, waarbij beperkt energieverbruik (gebruiksduur van de batterij) en beperkte afmetingen de belangrijkste randvoorwaarden zijn.

De 3 gezamenlijke eisen van weinig feedback, responsiviteit en eenvoud zijn vaak tegenstrijdig. Hierdoor is het ontwerp van een "behendige" transmissie strategie een lastige uitdaging. Vanwege de eis op de benodigde hoeveelheid feedback kan slechts beperkte CSI gebruikt worden. Het uitvoeren van een optimale strategie gebaseerd op deze beperkte kanaalkennis vereist gecompliceerdere bewerkingen dan wanneer het kanaal volledig en exact bekend is, omdat rekening gehouden moet worden met de

onzekerheid van de toestand van het kanaal. Daardoor wordt de transmissie strategie gecompliceerd en *niet* eenvoudig te implementeren. Evenzo kan een strategie gebaseerd op beperkte feedback minder responsief zijn, vanwege het ontbreken van exacte kanaalinformatie.

Veel moderne communicatiesystemen zijn pakket-geschakelde systemen. Deze systemen delen de te verzenden informatie op in blokken, pakketten genaamd, die apart van elkaar getransporteerd worden van zender naar ontvanger. Om betrouwbare, foutloze communicatie van alle pakketten te bewerkstelligen, worden in de verschillende protocollagen onafhankelijk de fundamentele problemen van communicatie over een onbetrouwbaar draadloos kanaal opgelost. In dit proefschrift beperken we ons tot de twee onderste lagen: de data link layer (DLL) en de fysieke laag (PHY). De Logical Link Control (LLC) laag, die de bovenste helft van de DLL vormt, houdt bij welke informatie bits verzonden zijn en verzorgt zo nodig eventuele hertransmissies. De bits worden dan doorgegeven aan de Medium Access Control (MAC) laag; de onderste helft van de LLC-laag. De MAC beslist over het moment van toegang tot het kanaal, met als doel het beperken van conflicten tussen pakketten van verschillende gebruikers. De PHY-laag, tenslotte, moduleert de informatiebits tot een fysiek signaal. Na verzending over het draadloze kanaal, wordt het signaal ter decodering doorgegeven aan de fysieke laag in de ontvanger. De informatie of het decoderen gelukt is, en, indien beschikbaar, de gedecodeerde bits worden vervolgens doorgegeven aan de hogere lagen.

In dit proefschrift ontwikkelen we *agile transmission strategies* voor de PHY, MAC en LLC lagen, met als doel het verhogen van de spectrale efficiëntie en de datasnelheid voor de gebruikers. We gaan hierbij uit van transmissie-adaptatie per pakket, om zo de responsiviteit te garanderen. We laten zien dat er zelfs met beperkte feedback een substantiële verhoging van de datasnelheid behaald kan worden ten opzichte van het geval van geen feedback. Bovendien blijkt het grootste deel van deze winst al te halen met eenvoudige adaptatie algoritmes, in plaats van de vaak zeer complexe optimale algoritmes. Om een effectieve marktintroductie mogelijk te maken, sluiten de in dit proefschrift voorgestelde strategieën aan op gangbare technologieën in bestaande standaarden. Zo introduceren we bijvoorbeeld ten behoeve van verbeterde adaptatie een nieuwe manier van gebruik van de acknowledgement (ACK) bits, zoals die al als feedback in gebruik zijn in de IEEE 802.11 standaard.

Het proefschrift begint met de bestudering van single-carrier systemen. In Hoofdstuk 2, bekijken we rate adaptatie in de LLC laag. Een eenvoudig ARQ schema levert beperkte CSI gebaseerd op ACK feedback. We stellen een eenvoudig adaptatieschema voor, dat een significante verbetering bewerkstelligt ten opzichte van een systeem zonder feedback. In Hoofdstuk 3 introduceren we een systeem met een meer geavanceerd ARQ schema gekoppeld met een rate adaptation algoritme. Ondanks een verbeterde performance is de implementatie van dit systeem nog steeds eenvoudig en blijft ook de feedback beperkt. Vervolgens worden multi-carrier systemen beschouwd. In Hoofdstuk 4, richten we ons op een pre-transformed orthogonal frequency division multiplexing (PT-OFDM) systeem. We introduceren een iteratief ontvangeralgoritme met lage implementatiecomplexiteit. De uitbreiding van PT-OFDM systemen met ARQ is

het onderwerp van Hoofdstuk 5, waar we een eenvoudig subcarrier-toewijzingsschema voorstellen. In Hoofdstuk 6 betrekken we ook de MAC-laag en kijken we naar multi-user communicatie. Het request-to-send (RTS) en clear-to-send (CTS) mechanisme wordt gebruikt om multi-user interferentie te beperken. We beschrijven een rate adaptation schema, gebaseerd op een nieuwe *successive capture analyse*, dat gebruik maakt van de RTSCTS signalering als (beperkte) feedback. Hoofdstuk 7 geeft tenslotte de conclusies van het proefschrift en beschrijft aanbevelingen voor verder onderzoek.

Acknowledgements

First and foremost, I would like to thank my promoters Prof. Jean-Paul Linnartz and Prof. Frans Willems for guiding me. I thank Prof. Sem Borst, Prof. Behrouz Farhang-Boroujeny, Prof Rob van der Mei, Dr. Job Oostveen and Dr Sumei Sun for being in the Doctorate Committee and for providing useful feedback on my dissertation.

Throughout my stay in Eindhoven, Jan provided continuous moral support and guidance on many fine aspects of research. I would like to thank Tjalling Tjalkens, Sjoerd Ypma, Yvonne Bokhoven, Yvonne Broers and Anja de Valk-Roulau for providing advice and support in TU/e. I also thank Tim Schenk, Jamal Riani, Steven van Beneden, Andrei Sazonov, Emanuel Habets, Rabotti Chiara, Tanya Ignatenko and many others for their valuable discussions and companionship.

I spent much time in Philips Research with many brilliant minds. I had exciting technical discussions with Job, Frans and Jean-Paul, as well as Stan Baggen, Ludo Tolhuizen, Ronald Rietman and Dee Denteneer. I also benefitted from the interesting viewpoints and companionship of Eric Penning, Sri Andari Husen, Hendra Gunawan, Admar Schoonen, Alessio Filippi, Semih Serbetli, Ashish Pandharipande, Wang Ying and Lorenzo Feri, among numerous others.

For friendship and for collaborations in papers to collaborations in cooking, I thank Ah Hung and Yuki, Hongming, Yuan Wei, Wang Qi, Huang Li and Cathy, Han Jungong and Chunmei, Cai Rong, Nianyong, Yu Yikun, Li Ping, Hu Hao, among others.

I would like to acknowledge the support from Sun Sumei and Francois Chin who made possible the transition from my post in Institute for Infocomm Research (I^2R) to the PhD position in TU/e, and eventually from TU/e back to I^2R.

The understanding and encouragement of my family has made everything worthwhile.

Curriculum vitae

Chin Keong Ho was born in Singapore in 1974. He received the B.Eng. (first-class Honors) and M. Eng degrees from the Department of Electrical Engineering, National University of Singapore in 1999 and 2001, respectively. He was with Institute for Infocomm Research, A*STAR, Singapore, since August 2000.

In October 2004, he took leave to work toward the Ph.D. degree at the Signal Processing Systems Group, Eindhoven University of Technology, The Netherlands. In his Ph.D. work he conducted joint work with Philips Research Laboratories, Eindhoven.

In May 2007, he returned to Institute for Infocomm Research as a Research Fellow. His research interest mainly lies in adaptive and cooperative wireless communications for multicarrier and space-time communications.