UNIVERSITÀ DEGLI STUDI DI PADOVA

Sede Amministrativa: Università degli Studi di Padova
Dipartimento di Ingegneria dell'Informazione

SCUOLA DI DOTTORATO DI RICERCA IN INGEGNERIA DELL'INFORMAZIONE
INDIRIZZO ELETTRONICA E TELECOMUNICAZIONI
CICLO XX

CROSS-LAYER DESIGN IN WIRELESS NETWORKS

Direttore della Scuola: Silvano Pupolin
Supervisore: Michele Zorzi

Dottorando: Simone Merlin

31 Gennaio 2008

Abstract

In this thesis, cross-layer optimization techniques for wireless networks are investigated. An introduction to the concept of cross-layer design is provided, reviewing the related literature, from both an architectural and an analytical point of view.

Three original contributions, which jointly address the optimization at different levels of the protocol stack are then presented.

The first contribution refers to a theoretical approach to channel allocation in multi-channel ad hoc networks, where each node is provided with multiple radio interfaces. An algorithm for the joint solution of congestion control, channel allocation and transmission scheduling is proposed.

The second contribution refers to a cross-layer optimization framework in the context of standard IEEE 802.11 WLAN. A mathematical model for the link performance is developed, and a sufficient description for the medium status is defined which allows to account for propagation and interference conditions. The optimization framework is used to develop algorithms for rate adaptation and VoIP quality enhancement which are adaptive to a broad range of working conditions.

Resource allocation in wireless cellular networks is also addressed and the problem of trading fairness for physical layer efficiency is investigated by means of a simple algorithm spanning PHY, MAC and LL layers.

In the end, additional published contributions related to the cross-layer paradigm are introduced, regarding microeconomic aspects in resource allocation and efficiency considerations about scatternet topologies in Bluetooth networks.

Sommario

In questa tesi viene discusso il concetto di *cross-layer design* in reti wireless. Il significato del termine cross-layer, in base alla vasta letteratura esistente, è introdotto sia dal punto di vista architetturale, che dal punto di vista dei tentativi teorici di formulare matematicamente il concetto. I contributi originali di questa tesi riguardano tre esempi di progettazione cross-layer in diversi ambiti applicativi.

Il primo contributo, di carattere teorico, è riferito a reti ad hoc in cui sono disponibili canali multipli e ciascun nodo della rete può essere provvisto di interfacce radio multiple. In questo ambito viene proposto uno studio di tipo analitico che porta alla definizione di un algoritmo per la risoluzione congiunta dei problemi di controllo di congestione, allocazione dei canali e scheduling delle trasmissioni.

Il secondo contributo riguarda la proposta di un meccanismo di ottimizzazione cross-layer nello specifico scenario di reti IEEE 802.11 standard. Viene presentato un modello matematico per le prestazioni dei link, basato sulla definizione di una descrizione sufficiente dello stato del mezzo. L'architettura proposta è successivamente usata per realizzare una algoritmo di rate adaptation e un algoritmo per l'ottimizzazione della qualità di connessioni VoIP.

Il terzo contributo fa riferimento ad una rete multi-cellulare basata su un accesso al mezzo di tipo FDMA. In tale scenario, viene discussa l'interazione fra l'allocazione delle risorse a livello fisico e la gestione della qualità del servizio a livello di link layer.

Vengono inoltre introdotti studi ulteriori oggetto di pubblicazioni internazionali, nei quali vengono toccati aspetti microeconomici nell'allocazione di risorse e considerazioni sull'efficienza di diverse topologie in scatternet Bluetooth.

Contents

List of Figures

Chapter 1

Introduction

Wireless and wireline communications have rapidly evolved in the last decade, following the increasing data rate and quality of service requirements. The increasing demand for multimedia contents such as video streaming, voice over IP services, gaming, and the massive peer-to-peer transfer phenomena, rises the need for very high performance communications networks.

The aforementioned applications do not only require high data rate communications, they also require the traffic to be delivered with additional constraints related to delay, reliability and security. For instance, audio or video streaming communications impose strict requirements in terms of delay and minimum guaranteed bandwidth. The delay issue is even more important in real time applications such as gaming or remote controlling. Peer-to-peer communications are instead delay tolerant, but they have very high and bursty bandwidth demand.

Moreover, the current trend is the design of communication networks able to concurrently support many different applications in a seamless way, also exploiting the convergence among different technologies. As a consequence, all different kinds of traffic, generated by different applications, have to be carried over the same network, sharing common resources. While this approach allows for an increased flexibility, it also rises huge problems in the network design, as it requires networks to be adaptive and reconfigurable in order to achieve optimum performance under different working conditions.

In particular, the packet switched architecture seems to be the preferred solution, and fixed and mobile services providing voice and multimedia communications are migrating toward this paradigm. This architecture allows for great flexibility, but also presents challenging and unresolved issues to be dealt with at all layers in the protocol stack. Efficient congestion control, adaptive routing algorithms and secure communications with guaranteed QoS are only some of the issues currently being investigated.

When considering in particular wireless and mobile networks, all the previous problems are exacerbated by the peculiar features and limitations imposed by the wireless channel. Both cellular and ah hoc networks have to cope with some basic issues such as scarcity of spectrum, mobility, and limited power supply. The scarcity of available spectrum, limited by the regulations or by the interference caused by preexisting technologies, requires the design of bandwidth efficient systems. Mobility requires transmission systems able to cope with the challenging propagation environment and protocols able to deal with changing topology. Limited power supply asks for efficiency at all layers of the protocol stack in order to reduce the energy consumption, while guaranteeing adequate performance.

From this brief overview it is clear how the design of network architectures and the optimization of their performance is a very challenging task, where many different requirements have to be satisfied.

The formerly proposed layered architecture based on the ISO/OSI reference model and the TCP/IP stack, though of fundamental importance in allowing for an easy design and for understanding the basic network functionalities, has been shown to impose great limitations on the performance optimization.

One of the basic concepts related to the layered architecture is the separation among different layers, i.e., a function realized inside a layer has to be performed independently of the specific implementation of all other layers. This assumption allows for the independent design of each layer. It is easy to show that this assumption does not properly model the real network behavior since, in actual networks, functions realized at different layers in the protocol stack interact with each other in a complex way, such that the layering model itself represents a fictitious simplification.

Moreover, due to lack in analytical modeling, most of the protocols complying with that architecture have been designed based on heuristic considerations and without accounting for the interaction with all the other protocols concurrently running at different layers. It happens thus that the joint behavior of all of them can lead the network to a suboptimal working condition, which sometimes can not even be predicted.

A well known and easy example is represented by the TCP mechanism, designed for congestion control at the transport layer in wireline networks, where the packet loss rate at the physical layer is typically negligible. Due to the lack of analytical models at the time TCP was designed, the mechanism has been based on heuristic considerations and its performance has been shown to be suboptimal. Moreover, when applied to wireless connections, TCP shows poor performance, since the higher packet loss ratio experienced in wireless channels is misinterpreted by the TCP mechanism, leading the connection to a wrong working point. Only recently, TCP has been reverse-engineered by using suitable analytical network models for the interaction between transport and physical layer, achieving a great performance improvement.

The investigation of the relationship among different layers and the design of mechanisms which break the classic layering, has lead to the concept of *cross-layer optimization*, CLO in the following. In this context, the study is devoted to understanding how different layers interact with each other and how this behavior can be modeled in a more efficient way than the current layered structure does.

Different approaches for designing cross-layersolutions have been taken, from both architectural and performance optimization perspectives. Heuristic solutions based on a redesign of the protocol stack have been proposed, which enhance the former architecture by allowing for a deeper interaction among different layers. The interaction among layers is achieved by using dedicated signaling channels, by merging different layers or by completely redesigning the system model.

The understanding of the complex interactions, and thus the correct design of new architectures, turns out to be very difficult. Nonetheless, some analytical approaches have been also investigated which start from mathematical modeling of the network and use this model to define new architectures. Some of them only refer to the study of the interaction among a limited number of layers, usually neighboring layers in the classic protocol stack. A plethora of proposals has been presented in the literature addressing specific issues in this context.

Recently, more exhaustive analysis have been proposed, which try to use mathematical argumentations for defining new concepts in the layering design.One of this approaches refers to the *layering as optimization decomposition*. In this case, the task of optimizing the network performance is mathematically decomposed in many subtasks which can be used to define a new layered architecture, where each layer is in charge of solving a specific optimization subtask, jointly working with all the other layers.

The concept of cross-layeroptimization is very general and can be applied to different network models. In this thesis, the focus in on wireless ad hoc and cellular networks and three main contributions are presented.

The first one is related to multi-channel multi-radio ad hoc networks, i.e., wireless ad hoc networks where each node is provided with multiple radio interfaces that can be tuned on different frequency channels. These kinds of networks inherit all the issues related to classic single channel ad hoc networks, plus some challenging problems due to the presence of multiple channels and multiple radio interfaces. The transmission scheduling becomes in fact more complex in particular when the number of interfaces is less than the number of channels. In this case the optimum binding of the interfaces to the transmission channels has in general a very high computational complexity and requires a centralized solver.

In this context a joint congestion control, channel allocation, and scheduling algorithm is derived by an analytical argument. The network is modeled as a flow graph introducing virtual links inside each node for the channel loading. The optimization approach follows the previously introduced concept of layering as optimization decomposition, and allows for the investigation of the network performance as a function of the number of channels, interfaces and concurrent transmission flows. This work has been developed during the visiting period at the University of Illinois at Urbana-Champaign and in collaboration with Prof. Nitin Vaidya.

The second contribution refers to the design of a cross-layer optimization framework for standard IEEE 802.11 networks. The standard provides a limited set of tunable parameters which can be set according to the designers' need. An optimization framework based on a mathematical modeling of the link performance is designed, which allows to tune such parameters according to different optimization goals. The mathematical model is based on the availability of a description for the status of the medium where the link is operating. A sufficient description for the medium status is thus defined and a way to estimate such status in real networks is proposed. Such an estimation is performed by using information available locally on each network card and explicitly described in the standard specifications. In particular we use the optimization framework to implement a rate adaptation algorithm for goodput optimization, GORA, and an algorithm for optimizing the perceived voice quality in a VoIP connection. Both algorithms exploit the mathematical model embedded in the framework and are able to adapt the network parameters in a broad range of working conditions, according to the specific optimization goal. The proposed mechanisms take into account the propagation characteristics and the congestion on the medium and set the parameters accordingly. In particular the rate adaptation algorithm is able to adapt the PHY rate with respect to both the channel impairment and the number of contending transmitter, which is one of the most original contributions. This work has been developed in collaboration with ST Microelectronics (Diego Melpignano, David Siorpaes) and other members of the SIGNET group at the University of Padua (Andrea Zanella, Federico Maguolo, Nicola Baldo).

The third contribution refers to multicellular wireless networks. The interaction between the resource allocation at the PHY-MAC layer and the scheduling at the LL layer is studied. The problem of defining a simple and distributed algorithm for the resource allocation (rate, power, subchannels allocation) which allows for an efficient use of the resources at the PHY layer is considered, together with a scheduling mechanism at the LL which is designed to provide fairness among different users. The mechanism is tested in an FDMA/TDMA multicellular scenario with a realistic channel model. A trade off between physical layer efficiency and quality of service provisioning arises and a tunable mechanism is discussed. This work has been carried out within the PRIMO project, granted by the Italian Research Ministry.

Additionally, further published contributions are briefly introduced in order to provide a broader overview on the possible applications of the cross-layerapproach. Performance optimization in Bluetooth networks is discussed and a microeconomic model is used to study the interaction between the resource allocation and the pricing mechanism in a wireless hotspot.

1.1 Structure of the Thesis

The thesis is organized in five additional chapters which are briefly described in this section.

Chapter 2 presents a more detailed introduction to the topic of this thesis while reviewing the literature. An overview of the cross-layer concept and the related different approaches, from both an architectural and a theoretical point of view, will be presented. In particular some proposed protocol stack architectures for communications among different layers, the concepts of *back-pressure* based scheduling and the paradigm of *layering as optimization decomposition* will be introduced. A detailed literature review will then be devoted to the specific topics that will be presented in Chapters 3, 4 and 5. In particular, multi-channel multi-radio ad hoc networks will be introduced, together with some modeling aspects, asymptotic results on capacity, theoretical approaches to performance optimization, and some specific issues on scheduling that will be useful in Chapter 3. Some practically oriented and heuristic solutions will also be listed. Resource allocation problems in cellular networks will then be presented, with particular attention to FDMA based networks for which the problem of finding efficient resource allocation algorithms at the MAC/PHY layer will be discussed.

Chapter 3 proposes an application of the cross-layerconcepts, describing an analytical model for multi-channel, multi-radio ad hoc networks. The proposed model is based on the recently developed concept of layering as optimization decomposition. An original algorithm for joint source rate adaptation, channel loading and scheduling is proposed. In this case, the layers in the new network architecture are redefined, based on mathematical considerations: each layer represents a different task in the optimization problem. The algorithm is described and simulation results are presented.

Chapter 4 deals with cross-layer optimization in IEEE 802.11 networks. The proposed optimization framework is introduced and the constituent blocks are described. In particular the definition of Medium Status is given, which is the basis for the development of the mathematical model describing the link performance. The framework is then applied for implementing two optimization algorithms. The first one refers to a Goodput Optimal Rate Adaptation (GORA). The algorithm is introduced and simulation results are presented which show the ability of GORA to adapt the PHY transmission rate to the propagation and the congestion conditions. The second one is focused on the optimization of a VoIP connection quality. The model for evaluating the voice quality and the resulting optimization algorithms are presented and evaluated.

Chapter 5 deals with multicellular networks and presents a mechanism for the interaction among the proposed physical layer resource allocator and LL scheduler. A simple architecture is introduced which allows to investigate the trade off between fairness in the provided service at the LL layer and efficiency at the PHY layer.

Chapter 6 presents additional published results related to cross-layerinteractions. In particular, the first proposed work investigates the interaction among the resource allocation at the MAC layer and the users demand by using a microeconomic model which accounts for the pricing mechanism applied by the provider. The second one is an analytical investigation on the efficiency of Bluetooth scatternet topologies, where the traffic pattern is also accounted for in the optimization.

Chapter 2

Related Work and Problem Statement

In this chapter, the basic knowledge needed for understanding the results presented in the next chapters will be provided, reviewing the work already appeared in the literature and introducing the models for the studies presented later on in this thesis. This chapter is organized in four parts.

In the first part, an introduction to the concept of cross-layering will be presented, showing some approaches appeared in the literature, from both a pure architectural and a theoretical point of view. Particular emphasis will be reserved to the analytical approach based on the concept of *back-pressure* and the paradigm of *layering as optimization decomposition*, that will be used in Chapter 3.

In the second part, the specific case of ad hoc networks is considered, with particular attention to the case of multi-channel multi-radio ad hoc networks, which clearly includes the case of classic single channel networks. Commonly used models for topology and interference representation are introduced, and some asymptotic results on capacity are reported. Different optimization approaches are presented which are based on the previously introduced network model. The problem of scheduling in such kinds of networks is introduced and some results are shown. Some heuristic approaches are finally reported, which are close to practical implementation.

In the third part, existing rate adaptation algorithms for IEEE 802.11 networks are briefly described in order to allow for a comparison with the peculiar features of the rate adaptation algorithm presented in Chapter 4. A model for quality evaluation of VoIP connections is also introduced.

In the fourth part, the case of resource allocation in cellular networks is considered. The problem of allocating physical resources in an FDMA/TDMA multicellular environment is introduced, together with the concepts of multiuser, frequency and time diversity. The benefit of cross-layer approaches is depicted, introducing the topic of Chapter 5.

2.1 CLO: general concepts

For many years, network design has been based on the well known ISO/OSI reference model or, more realistically, on the TCP/IP reference model. Such models have been of great importance in allowing for a clean and easy protocol design, facilitating the protocol standardization and thus the interoperability among different actual networks. Figure 2.1 shows the two famous networking models.

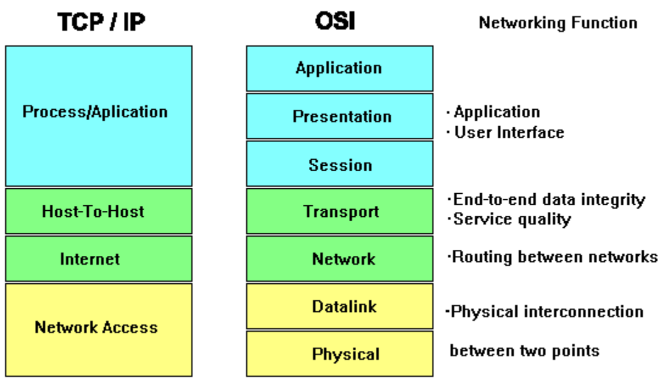

Figure 2.1: ISO/OSI and TCP/IP protocol stacks

It is worth recalling the principles which inspired the proposal of the layered architecture.

- A new layer has to be created where a different abstraction is needed;
- each layer should perform a well defined function;
- information flow across-layers should be minimized;
- the number of layers should be large enough such that different functions are in different layers;
- the function of each layer should be easily mapped on a protocol.

These rules essentially state a hard separation among the functions inside different layers, as each layer has to appear as a black box to all the others.

This model allows for an easy protocol design, as each protocol resides inside a single layer and has to realize only the functions for which that layer is in charge. Moreover the interaction with higher and lower layers is limited to the knowledge of the general input/output specifications provided by the interface used to connect with neighboring layers, thus each protocol can be implemented independently of all the other ones.

Nonetheless, recently the need for pushing the network performance toward its limit has revealed the intrinsic limitations imposed by these layering assumptions.

Some authors note that layering is itself an artifact because of the actual complex interactions existing among all different networking functions [5]. Thus, the model needs to be reconsidered for a better understanding and optimization of the network behavior.

In particular, a deeper interaction among neighboring and non-neighboring layers seems to be the fist step for enhancing the network performance, thus the term cross-layer design. More effective solutions could also require to completely break the layered model or use a different concept of layering.

The idea of cross-layer design in networking is old [6], though there is not yet a clear definition and a unique approach. The main reason for this lack is related to the complexity of the network structure, which has not yet been fully understood and modeled from a mathematical point of view. Clearly, form a practical perspective, changing the network architecture model would bring enormous problems with existing and deployed architectures and protocols. That is why pure cross-layer approaches are mostly confined to the literature, while practical solutions still have to cope with the layered architecture and compromise solutions have to be considered.

Nonetheless, different proposal for enhancing the well known ISO/OSI and TCP/IP stacks have appeared in the literature and a plethora of specific work which breaks the layered architecture with the aim of improving the network performance has been published. A first approach is shown in Figure 2.2, taken from [3]. Here the focus is on specific inter-

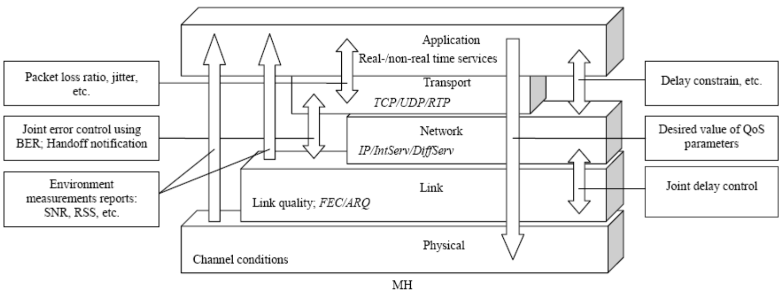

Figure 2.2: Cross layer architecture [3]

actions among layers, and on the needed signaling messages. In particular each layer needs to exchange control information only with a subset of the remaining layers and the correspondent message function is also introduced. In particular, note the interaction between the application and the physical layer which can be used to set the user's need according to the PHY bandwidth and viceversa.

A more general concept refers to the enhancement of the existing layered architecture, by defining general methods for exchanging control messages among different layers. As an example, Figure 2.3 reports the proposal appeared in [4], where the classic protocol stack is augmented by superimposing transversal control planes. Each control plane is in charge of a different function and can interact with all the layers in order to achieve its optimization goal. Thus each plane acts both as a communication and a control plane.

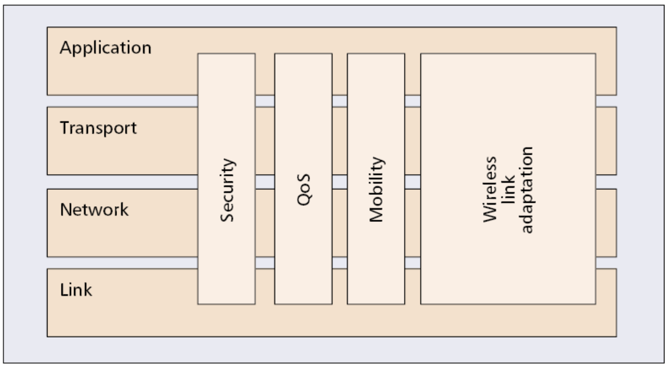

Figure 2.3: Cross layer architecture [4]

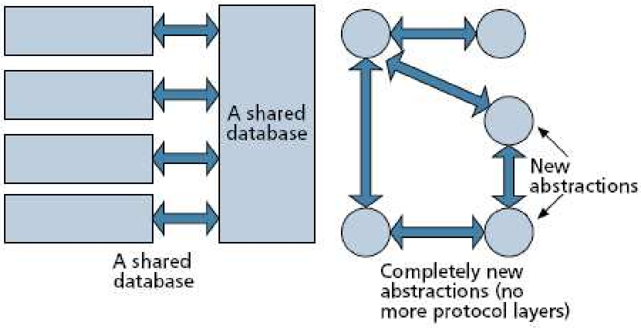

Figure 2.4: Cross layer architecture

A further general approach is in Figure 2.4. In this proposal, all layers communicate with a single control plane, which is devoted to controlling all the layers functions in a unified way, according to some optimization criteria. In this case the control plane, which becomes the core of the network node, can actually be used to create a new abstraction of the network functionalities, and thus the layered structure loses most of the original meaning.

Many specific solutions have been proposed dealing with the interaction among a limited number of layers. For the sake of providing some simple proof-of-concept, in the following a few simple and well known examples are reported in order to clarify the concept of cross-layer interaction and show the potential benefits.

A simple example of interaction between the functions of the LL layer and the PHY layer is referred to the case of the LL layer being able to exploit multiuser diversity, which is a property related to the physical layer behavior [7]. Consider an access point using a TDMA MAC with three users connected to it. Only downlink transmissions are considered and each channel between the access point and a user has equiprobable ON/OFF states due to fading phenomena. In case the users are served by a pure channel unaware round robin mechanism, the effective channel utilization given to each user is 1/6. Using a round robin mechanism which only serves users with a good channel state, the overall channel utilization grows to 7/8, and the per–user available channel is 7/24, which almost twice than in the channel unaware case. Thus the interaction between PHY and LL layers can be beneficial.

An example of the need to break the layered structure is also represented by the interaction between transport layer and physical layer in a wireless system. TCP is a congestion control mechanism which adapts the traffic injected into the network, based on measurements of the packet losses, which is assumed to be an indication of the link congestion. In a wireless link, this assumption does not hold anymore, due to the high probability of a packet loss being caused by a channel impairment.

For a correct behavior of the TCP mechanism, the two causes of packet losses should be discerned. Two proposed solutions refer to the Snoop TCP and the TCP Freeze [8, 9]. In the former one the packet losses occurring within the wireless link are hidden to the TCP mechanism by adding a packet cache at the IP layer and retransmitting the packets only within the wireless link, without notifying the TCP layer (thus breaking the end-to-end nature of TCP). In the latter one the receiver can block the connection by setting a null receiving window as long as the channel is in a bad condition (thus enabling a channel aware behavior).

As pointed out in [10], it may also happen that unintended cross-layer interactions can have undesirable consequences on overall system performance. Once the entire network stack is considered, cross-layer design may lead to cycles in the logical architecture, since many interactions are not easily foreseen.

Moreover, the real power of modularity may be lost. That is, for the sake of optimizing performance, cross-layer techniques make vain the possibility of designing protocols at a particular layer without worrying about the rest of the stack.

A first example of bad CL design in [10] is based on a 802.11 scheme using rate adaptation, showing that paths found by any popular DSDV implementation may prove to be highly inefficient due to rate adaptation. Also, even stability may be an issue, and a second example, shows that when the rate is adapted to the signal to interference ratio the interaction may bring networks to be unstable.

All the above described approaches and considerations on cross-layering are based on pre–existing layered structures and make use of intuitive, heuristic or trial–and–error considerations which need to be designed and verified case-by-case.

Clearly, the availability of a complete mathematical model for the network behavior would solve the problem by allowing for a reliable definition of the cross-layering concept and a systematic design of performance enhancing solutions.

Recently, some mathematical models have been presented, which are based on simplified scenarios but nonetheless represent a promising way. The next subsection is devoted to the presentation of such mathematical formalization.

2.2 CLO: analytical aspects

In this section the focus is on cross-layer solutions based on a theoretical background.

A seminal work introducing a control theory approach to network optimization is presented in [11], where algorithms for congestion control at the transport layer has been introduced and the problem of regulating the source traffic injected into the network is solved by using control theory. Sources compute the optimum flow rate based on a feedback price accounting for the network load and by using an iterative algorithm, which is proved to converge to the optimum solution under certain assumptions.

Algorithms for joint congestion control and transmission scheduling have then been proposed [12] which are able to jointly optimize source rate, link scheduling and routing [13, 14, 15] including also the power control operation [16, 17].

The mathematical tools widely used in these analytical approaches are optimization problem decomposition by Lagrange relaxation, sub gradient algorithms and Lyapunov stability [18, 19]. In the following, a detailed description of a general mathematical formulation for designing cross-layer algorithms is presented. This formulation can be used to model the network at different layers and is suited for designing cross-layer optimization algorithms. It will be used in Chapter 3, for developing a joint congestion control, channel allocation and scheduling mechanism for multi-channel multi-radio ad hoc networks.

The concept of "layering as optimization decomposition" has been investigated in the last few years as a powerful way for analytically defining cross-layer optimization problems and at the same time designing feasible algorithms for their solution [20].

In order to describe the analytical formulation of the concept, some notation is introduced. The set of nodes is $\{n : n = 1, \ldots, N\}$, with N the number of nodes. Traffic flows are, in general, carried over multi-hop routes. Each end-to-end unicast connection will be referred to as a *commodity* in the following. Let $\{s : s = 1, \ldots, S\}$ be the commodities set, where s can be considered as the index representing all the flows going toward a same given destination. Note that a commodity is characterized by the destination node, so that multiple sources with the same destination can be considered as a single commodity. The input rate for commodity s at node n is λ_n^s.

Let $\overline{\lambda}$ be the vector of all input rates. Each input rate can assume values $\lambda_n^s \in \Lambda_n^s$. Each node n will be provided with an input queue U_n^s for each commodity s All the incoming traffic for commodity s is loaded on queue U_n^s. Let $r_{a,b}^s$ be the transmission rate associated with the flow between nodes a and b carrying traffic for the commodity s and $\overline{r}$ be the corresponding vector for all a,b,s. The physical layer capacity for the link between nodes a and b is denoted as $w_{a,b}$. Let us denote by $\overline{w}$ the vector consisting of $w_{a,b}$ for all nodes a, b.The feasible rate region, i.e. the set of all feasible $\overline{w}$ vectors, is denoted as $\mathcal{W}$, which depends on the interference model. A general communication and interference model is initially assumed, which can be used to model both ad hoc and cellular networks, being the formulation independent from the specific communication model.

The utility function for commodity s associated with each source node n is denoted by $G_n^s(\lambda_n^s)$. To allow the use of convex optimization techniques, all the utility functions are assumed to be strictly concave, and the rate vectors $\overline{w}$ will actually be considered as belonging to the convex hull of the set $\mathcal{W}$, $\overline{w} \in \mathrm{Co}(\mathcal{W})$. Similar assumptions have been made in past work as well [14, 20]. Different utility functions can be used to achieve different network behaviors.

Define the *stability* of a network as the condition of having limited queue length, more precisely $\lim_{t\to\infty} E[\sum_{n,s} U_n^s] < +\infty$, where $E[\cdot]$ represents the expectation operator.

Let Λ be the *capacity region* of a network, i.e., the set of all feasible input rate vectors $\overline{\lambda}$ for which the network is stable.

In the following, a general formulation is presented in terms of an optimization problem on a network flow where the goal is the utility maximization. The problem is then decomposed by using a Lagrangian relaxation. This operation will allow to define the optimization layers and a distributed algorithm for the utility maximization as proposed in [20]. The algorithm turns out to be based on the concept of "back pressure" scheduling [14]. Similar solutions are presented in [13].

The proposed examples will show a cross layer algorithm which allows for the interaction between a congestion control mechanism at the transport layer, the scheduling at the LL layer and the resource allocation at the MAC/PHY layers.

More complex algorithms including further optimizations can be derived using the same approach as will be shown in Chapter 3. A comprehensive study about different way of decomposing the optimization problem is presented in [21].

In the following, as an example, two decompositions are shown which are usually referred to as "link centric" and "node centric".

2.2.1 General problem formulation

Based on the previously introduced notation, the goal is to solve the following optimization problem:

$$\max_{\overline{\lambda},\overline{r},\overline{w}} \sum_{n,s} G_n^s(\lambda_n^s) \tag{2.1}$$

s.t.:

$$\sum_i r_{i,n}^s + \lambda_n^s \leq \sum_j r_{n,j}^s \;\forall n, s \tag{2.2}$$

$$\sum_s r_{i,n}^s \leq w_{i,n} \;\forall i, n \tag{2.3}$$

$$\overline{w} \in Co(\mathcal{W}) \tag{2.4}$$

$$\lambda_n^s \in \Lambda_n^s \;\forall n, s \tag{2.5}$$

In the previous model:

- (2.1) is the objective function
- (2.2) is the flow conservation constraint at each node
- (2.3) is the constraint that the aggregate flow on a link must be less than the physical rate
- (2.4) is the feasible rate region for the actual links.
- (2.5) is the feasible set for the input rates.

Symbols:	
$G_n^s(\lambda_n^s)$	Utility function
$\overline{\lambda} = [\lambda_n^s]$	Injected input rate
$\overline{r} = [r_{a,b}^s]$	Flows associated to channel-link-commodity connections
$\overline{w} = [w_{a,b}]$	Physical rates associated to physical channel-link
$\mathcal{W}$	Feasible rate region for actual physical links
Λ_n^s	Feasible input rates

Table 2.1: Symbols

Based on the assumption about the utility functions and on the convexity of the domain, (2.1)-(2.5) is a convex optimization problem [12, 21, 15].

2.2.2 Node centric solution

In the node centric approach (also referred to as "route independent") the solution to the optimization problem is obtained via its dual problem, relaxing all the constraints (2.2).

Let $\mathbf{U} = [U_n^s]$ be the vectors for all the Lagrange multipliers associated to constraints (2.2). Relaxing the constraints (2.2) and (2.3), the Lagrange dual function for the problem is:

$$L(\mathbf{U}) = \max_{\overline{\lambda},\overline{r},\overline{w}} \left\{ \sum_{n,s} G_n^s(\lambda_n^s) + \sum_{n,s} U_n^s \left(\sum_j r_{n,j,c}^s - \sum_{j,c} r_{j,n}^s - \lambda_n^s \right) \right\},$$

where the optimization variables $\overline{\lambda}, \overline{r}, \overline{w}$ are still subject to constraints (2.3)-(2.5).

The previous expression can be rewritten as:

$$L(\mathbf{U}) = \max_{\overline{\lambda}} \left\{ \sum_{n,s} G_n^s(\lambda_n^s) - \lambda_n^s U_{n,in}^s \right\} + \tag{2.6}$$

$$+ \max_{\overline{r},\overline{w}} \left\{ \sum_{i,j,s} \left(U_i^s - U_j^s \right) r_{i,j}^s \right\} \tag{2.7}$$

$$\tag{2.8}$$

Note how each maximization represents a different "layer" in the optimization task:

- (2.6) congestion control;
- (2.7) flow allocation, routing and physical rate allocation.

The two layers uses $\mathbf{U}$ as coupling variables.

The solution of the dual problem requires the computation of the $\min_{\mathbf{U}} L(\mathbf{U})$. In case the starting optimization problem is convex, then the solution of the dual problem yields the solution of the original optimization problem [22] (there is no duality gap).

Such minimization can be performed by using a subgradient based algorithm, which is used to iteratively search for the U_n^s. If we identify each iteration of the subgradient algorithm as a different time slot t, then the updating equation for the U_n^s is

$$\begin{aligned} U_n^s(t+1) = \Big[U_n^s(t) + \alpha(\tilde{\gamma}_{n,c}^s(\mathbf{U}(t)) + \\ - \sum_j \tilde{r}_{n,j}^s(\mathbf{U}(t))) \Big]^+ . \end{aligned} \tag{2.9}$$

The described algorithm allows for exploiting the full capacity region available in the network, under certain hypothesis on the α value.

From the structure of the equation (2.9), in case $\alpha = 1$, $U_n^s(t+1)$ represents the queue length at node n for traffic of commodity s. This property has already been pointed out in [15, 13, 23].

The most challenging issue in applying the algorithm to real networks is the fact that the scheduling related maximization requires the knowledge of the feasible rate region.

2.2.3 Backpressure

Similar results to the ones shown for the node centric case are in [14], where the problem is approached from a control theory perspective.

The core of the algorithm presented in [14] is based on the same maximization used in the second line of equation (2.7). The quantity involved in that maximization $\left(U_i^s - U_j^s\right) r_{i,j}^s$ is referred to as "backpressure", since it accounts for the difference of the queues length at the input and output of each link. The maximization is such that the scheduled links are the one which carry the traffic in the direction of decreasing queues length.

In [14] the considered network physical layer assumes the network topology (and thus the channel state) can have a finite number of states, the time is slotted and within each slot the network status (topology and channel) is constant. The evolution of the topology in subsequents slots follows a Markov process. Under this assumption, the proposed algorithm is able to maximize the utility function to a value which is close to the optimal one, while stabilizing the network. The algorithm is provided with tunable parameters which can be used for trading optimality with stationary queues length (i.e. trading utility for delay)

2.2.4 Link centric solution

A different approach to the optimization decomposition refer to the "link centric" case. This formulation and similar ones, have been used for early study in congestion control [11, 18].

Indicate with l the generic link between nodes i and j. In the link centric approach, the additional constraint $\lambda_n^s = r_l^s \;\; \forall l \in L_s$ is considered, where L_s represents a precomputed path for commodity s. This assumption implies the input rate is simultaneously applied to all the links traversed by the flow.

Relaxing the constraint (2.3)

$$L(\mathbf{U}) = \max_{\overline{\lambda},\overline{r},\overline{w}} \left\{ \sum_{n,s} G_n^s(\lambda_n^s) + \right.$$
$$\left. + \sum_{l,s} U_l \left(\sum_{s:l \in L_s} \lambda^s - w_l \right) \right\},$$

The previous expression can be rewritten as:

$$L(\mathbf{U}) = \max_{\overline{\lambda}} \left\{ \sum_{n,s} G^s(\lambda^s) - \lambda^s \sum_{l \; inL_s} U_l \right\} + \sum_l U_l w_l \tag{2.10}$$

thus showing how the maximization requires two separate operations, one of them solving the congestion control and the other one solving the scheduling problem.

Similarly to the node centric case, the updating equation for the Lagrange multipliers turns out to be

$$U_l(t+1) = \left[U_l(t) + \alpha \left(\sum_{s \in i} r_l - w_j \right) \right]^+$$

and thus U represents queues at the input of each link.

It can be noted how the congestion controller reacts to the sum of the queues along the path. Each flow is associated with a predetermined path and it is assumed that the rate computed by the congestion controller is applied simultaneously to all the links. The scheduler is the throughput optimal one [14].

2.2.5 Non–ideality issues

Previously presented results consider the scheduling algorithm is provided with a perfect knowledge of the feasible rate region and is able to take optimum choices in the maximization of the scheduling metric (Equation 2.7).

In [23], the 'link centric approach' is extended by considering the case where an imperfect scheduler is used.

A crucial point is discussed. Let suppose the scheduling algorithm is not able to solve the exact scheduling problem of the backpressure maximization, then how does the congestion control behave? A generic scheduling algorithm is considered, which is only guaranteed to solve the scheduling function in an approximate way on each slot, within a scaling factor $\gamma < 1$ from the optimum one. In the paper it is argued that the joint rate control and scheduling algorithm may not converge (because of loops) or may converge to a non optimal point (with respect to the reduced capacity region). This implies a big limitation in the application of a joint congestion control and suboptimal scheduling. In the link centric case, it is proved that, using logarithmic utility functions, if it converges, then the solution will be "not too far" from the reduced optimal one, but still suboptimal.

This behavior changes if a stochastic model for the number of users is considered. In this case, the rate control converges to the optimum solution referred to a γ-scaled capacity region, whatever suboptimal scheduling policy is used.

On the contrary, using a "node centric" approach it is possible to use a γ-imperfect scheduler in order to obtain a γ-reduced maximum utility function [14] even in the case of static number of users.

The backpressure algorithm is in fact proved to converge even with an imperfect scheduler. If the maximization of the scheduling function is performed with a loss of at most γ then the scheduling policy is said to belong to the class of S_γ and the stability is guaranteed inside a γ reduced capacity region. This opens the way for the use of approximate algorithms, as described in the following.

In [24] the behavior of the classic primal-dual control algorithm is studied , for the case of a time varying capacity of the links. The novelty is that each link has a capacity $w_l(t)$ instead of w_l, thus depending on time. A continuous time system is considered. If there is no delay in the data transmission and on the price feedback, then the system is stable even with time varying channels. The stability is here defined as a "trajectory stability" since the equilibrium point changes as the channel condition changes.

If there is a delay in data transmission or price feedback, then the system is linearized around an equilibrium point and it is provided a set of conditions on the algorithm parameters that allow for local stability around the given equilibrium point. Such conditions turn out to be a function of the delay (a similar conclusion is derived in [18] (Chapter 6) for the case of time-invariant channels).

In all the previous proposals congestion control is performed only at the source node.

The paper in [25] explicitly addresses the problem of hop-by-hop congestion control, in a node exclusive interference model (the impossibility for each node to transmit and receive at the same time).

The proposed algorithm requires that each node along the data path performs a congestion control based on the price received from the next downstream node down the path toward the destination. Moreover each node passes the aggregated price to the upstream node, until the source is reached. Thus, eventually the source receives an aggregated price (similar to an end-to-end control, the difference is that the prices of all nodes are aggregated in a single message along the path) but each node performs congestion control (in a hop-by-hop manner). The advantage of the hop-by-hop approach is that the queues length is balanced and the bottleneck queue length is reduced (spatial spreading). Thus the algorithm uses essentially an end-to-end approach, repeated at each node along the path.

2.2.6 Scheduling

Interference models

The previously described algorithm for joint congestion control and scheduling requires the solution of the scheduling maximization problem (i.e. Equation 2.7), whose complexity depends on the feasible rate region structure. Here are described two commonly used interference models that will be used in the thesis.

Two interference models have been defined in [26]: i) Protocol model, ii) Physical model. Such models are widely referred in the literature and are used in many models. Both are binary interference models, i.e., they only state if a transmission will be correctly received or not and represent a rough estimation of the actual physical behavior.

In the protocol model, each node is associated a transmission radius r, such that only nodes within that radius can receive the transmission. A transmission is correctly received only if no other nodes transmit at the same time within a distance $r + \Delta r$ from the receiver. A variation to this model states that a transmission is correctly received only if no other nodes transmit within an interfering distance both of the receiver and the transmitter. The protocol model gives a geometrical based interference definition which is appropriate for graph based system modeling.

The physical model accounts for the SNIR at the receiver, which is calculated as the ratio between the useful received power and the sum of noise and interference power. Once fixed a target SNIR for the correct reception, the model turns out to be very similar to the protocol one. Some of the works proposed in the following, are valid both under the protocol and physical interference models. Anyway most of them provide explicit results only for the protocol model.

Solutions with Maximal Weighted Independent Sets

In case the protocol model is used for representing the interference scenario, each link l interferes with $\xi(l)$ neighboring links and the optimization required for the solution of the scheduling problem becomes a combinatorial problem.

In particular the problem of maximizing the back pressure function becomes a Weighted Maximum Independent Set problem. The problem is in general NP-HARD [27] and moreover no constant factor polynomial time approximation exists. Even the problem of finding distributed constant factor approximation for particular topologies is an open issue.

Clearly a greedy centralized algorithm which selects at each step the link with the highest metric and discards all the interfering links can achieve a reduced capacity region by a factor of $1/K$ where K is the maximum number of interfering nodes for each node [23], which is a trivial lower bound. In the case of multichannel networks with reduced number of interfaces the lower bound becomes $1/(K+2)$ [28]. Note that all the previous bounds are lower bounds and are usually very conservative. In [29] it is pointed out that such a greedy approach is optimal in graphs with particular structure. All the interfering graphs which satisfy a property called Local Pooling allow the greedy solution to be optimal. Among the interfering graph structures which satisfy the local pooling are trees, cliques and cliques interconnected by disjoint links.

Note that for the case of a node exclusive interference (i.e. no interference occurs among links and the only constraint is that no node can receive and transmit at the same time), the problem becomes a Weighted Maximum Matching which instead can be solved in polynomial time by a centralized algorithm [27]. In this case a distributed algorithm for weighted matching is known to achieve a 0.5 approximation of the optimal value [30].

A novel approach for the same problem is also in [31], where an iterative distributed algorithm is designed to approximate the maximum matching within an arbitrary constant factor. On each slot a new schedule is constructed in a distributed way. The new schedule is mixed with the one in the previous timeslot in order to obtain a new matching. The mixing procedure is based on the comparison of weight of the two matching. The mixing procedure can be performed using a distributed and iterative gossiping mechanism which can approximate the optimal solution with the desired precision (at the cost of a longer convergence time). The proposed algorithm can approximate a maximum matching (an extension to a weighted independent set is also claimed but not sufficiently investigated)

Another possible extension to the case of K-degree interference model is in [32]

Algorithms based on Maximal Independent Sets (non weighted)

The problem with the maximization of the back pressure function in the protocol interference model case is that a weighted maximization is required. Looking for different algorithms, which do not require a maximization in order to achieve the capacity region has lead to satisfying results only for single hop networks. Moreover, such algorithms have not been integrated with the congestion control and are mainly designed for single hop networks. In the following, some details are provided for this particular class of algorithms.

If the network is single hop, it can be stabilized by any random Maximal Independent Set scheduling which scheduled only backlogged queues. If the definition of backlogged queue U is $U > w_l$, with w_l the number of bits that can be served in a slot, then any random maximal schedule sequence allows for a capacity region which is $\sum_{j\in\eta(i)} \frac{\lambda_j}{w_j} < 1$, where

λ_j is the input rate and $\eta(j)$ is the set of interfering queues [33, 34].

In [28] an extension to a multi channel case is presented. The multi channel maximal independent set is defined as: either a link is scheduled, or one of its interferer is scheduled or no one of the interfering links has an available interface. Using this definition they propose an algorithm for loading the channels in such a way to keep the networks stable within a region that is $1/(K + 2)$ away from the optimal solution.

The extension to the multi hop case requires additional information at each node. A "regulated" maximal independent set is shown to allow for stability in a network where the routing is provided in advance and each node has the knowledge of the input rate. In this case each node is provided with an input queue and an output queue for each flow. The traffic from the input queue to the output queue is limited to be just slightly higher than the source rate. This brings back the multi-hop network to the well-behaved single hop one [33].

A different approach requires each node to keep track of the number of hops of each packet. Let q^k be the queue length accounting only for the packets which experienced k or less hops. Then an iterative algorithm can be used. At each step i a maximal independent scheduling can be applied only on the q^i backlogged queues, i.e., containing packets with a number of hops less than or equal to i. The matched links are added, all the interfering ones are removed and the iteration are repeated. Note that the complexity depends on the number of hops [35].

2.3 Ad hoc networks

Results for the specific case of ad hoc networks are reported in this section. Asymptotic results for the network capacity are followed by optimization approaches which do not follow the cross-layer optimization framework described in Section 2.2. Particular attention is devoted to the case of multi-channel multi-radio networks. Protocols mostly based on heuristic considerations are also presented.

2.3.1 Asymptotic analysis

The seminal work of Gupta-Kumar [26] has derived the asymptotic (in the number of nodes) capacity of multi-hop ad hoc networks, where each node is equipped with a single network interface card (NIC in the following), and a single channel with capacity W is available. n nodes are randomly scattered on a sphere surface of unit area. Results are valid both under the protocol and physical interference models.

Each node is assumed to have a random destination node for its transmissions. The asymptotic average per node throughput is derived. It is obtained as the union of a lower and an upper bounds computed using geometrical considerations. A lower bound is derived assuming a shortest-path-like routing and defining a feasible interference free transmission scheduling. The upper bound is derived based on connectivity arguments and on spatial channel reuse considerations. The two bounds lead to the following per-node throughput scaling law:

$$\lambda(n) = \Theta\left(\frac{W}{\Delta^2\sqrt{n\log n}}\right),$$

where Θ indicates that the actual asymptotic value is scaled by a constant.

Following the same reasoning, and adding some particular consideration for the multi-channel multi-rate case, Vaidya in [36] has shown how the capacity of a random ad hoc network scales with the number of nodes n, number of channels c, and number of interfaces per node m. Each interface is a half duplex interface and can be tuned to any of the c channels.

Even in this case, an upper and a lower bound have been derived. Each of the two bounds is actually the intersection of three bounds, each of them accounting for different performance limitation factors.

In the upper bound construction, three sources of capacity limitations are pointed out: connectivity, interference, and bottleneck nodes. To each of the three phenomena is associated an upper bound for the achievable capacity. The total upper bound is the intersection of the three.

$$\lambda(n) = \begin{cases} \Theta\left(\sqrt{\frac{W}{n\log n}}\right) & \text{if} \quad \frac{c}{m} = O(\log n) \\ \Theta\left(W\sqrt{\frac{m}{nc}}\right) & \text{if} \quad \frac{c}{m} = \Omega(\log n) \text{ and} \\ & \quad \frac{c}{m} = O\left(n\left(\frac{\log\log n}{\log n}\right)^2\right) \\ \Theta\left(\frac{Wm\log\log n}{c\log n}\right) & \text{if} \quad \frac{c}{m} = \Omega\left(n\left(\frac{\log\log n}{\log n}\right)^2\right) \end{cases}$$

It is important to note that until $\frac{c}{m} = O(\log n)$, the throughput scaling law is the same as the single channel and single radio case.

This implies that even if nodes are provided with a number of NICs less than the number of available channels the network can achieve the full capacity. We observe that in the proposed model $O(\log n)$ is a measure of the number of neighboring nodes.

This is due to the fact that in this region the capacity bound depends on channel occupancy. Channels are already fully utilized, thus the number of interfaces is not a bottleneck.

This implies for instance that in a network where each node is equipped with a single NIC, up to $O(\log n)$ channels can be fully utilized.

2.3.2 Optimization problems

All the proposed models deal with protocol or physical interference models. Since such interference models provide a binary connectivity and interference representation, the use of graphs is a natural way to deal with them. Here it is detailed the construction of connectivity and conflict (interference) graphs which is valid throughout almost all algorithms described in the following. Variations to the used models will be explicitly reported.

A connectivity graph is defined as a graph whose vertices are the nodes in the network. An edge is placed between two nodes if they are within communication range.

A conflict graph is consequently defined to account for interference issues. In this graph, vertices's represent the communication links (i.e., the edges of the connectivity graph) and edges are placed between vertices's corresponding to interfering links, according to the protocol model. It is worth recalling that the interference range can be greater than the communication range.

The optimization problems here described are formalized as linear on non-linear (integer) programming problems. The commonly considered constraints are given by a subset of the following ones: i) connectivity; ii) interference; iii) number of available channels; iv) number of available radios; v) traffic demand; vi) fairness in the offered traffic.

Different proposed models account in different ways for the above constraints. Particular interference and connectivity measures are also defined. Some models embed constraints on the connectivity and interference directly in the graph structure. In such a case additional vertices's or edges are introduced. This will be specified in the algorithm descriptions.

The main optimization goals refer to i) interference minimization or ii) throughput maximization and they are pursued optimizing appropriately defined metrics.

Another classification refers to algorithms which account for i) a packet by packet channel allocation (thus solving also the related time scheduling problem) and ii) algorithms which account for long time scale allocation steps. In the former case the goal is to prevent two interfering nodes from being on the same channel at the same time (this could be the case for synchronous TDMA systems). In the latter one two interfering nodes are assumed to share a channel in the long term. This is the case of a CSMA/CA MAC protocol where the transmission scheduling in time is implicitly solved by the MAC mechanism.

Channel allocation

The approaches listed in this subsection refer to optimization problems where constraints related to the traffic demands (constraints v) and vi)) are not considered. Equivalently, such constraints are assumed to be solved a priori.

The optimization deals with NICs and channels allocation. The objectives can be interference reduction (different metrics for interference evaluation are proposed) or throughput maximization (different metrics for throughput evaluation are proposed)

A common assumption is that nodes are in saturated condition, thus the allocation of a channel to a NIC leads to its occupation by a tentative transmission.

In the paper in [37] the protocol interference model is enhanced by introducing weights on the conflict graph edges. The interference at a vertex in the conflict graph is defined as the sum of the weight of incident edges. Thus an optimization problem is defined which aims at minimizing the maximum interference value on each link, satisfying the connectivity, radio and channel usage constraints. Only a centralized heuristic based on graph coloring (CLICA: Connected Low Interference Channel Assignment) solution is provided. Clearly this approach is suitable for a CSMA MAC. Results are provided in terms of the interference metric above defined. Results in ns2 based simulations of a 802.11 system are also provided in terms of throughput.

A similar approach is in [38] where the classic protocol model is used and the objective is to minimize the maximum number of interfering edges incident on a communication edge (vertex of the conflict graph). The problem is shown to be similar to a max k-cut in a graph with the additional constraint of the card number. Two heuristics, one based on TABU graph coloring and the other one based on a greedy approach are presented. Moreover the problem is formalized as an ILP and two relaxations are presented in order to find lower bounds on the minimum interference value. It is also shown how non uniform traffic requirements and non orthogonal channels can be considered appropriately defining the interference function. All the proposed solutions are compared with the CLICA [37] both with graph theoretical metrics and ns2 802.11 throughput simulations.

In [39] the problem is formalized as an ILP in two different ways. In the first one, the optimization objective is the maximization of the number of concurrent active links which satisfy the constraints on interference, number of channels and number of NICs. Some additional constraints are added for the case of LP relaxation. In the second one, the connectivity graph is modified substituting each communication edge with a number of communication edges equal to the number of channels. Each link is also weighted with the expected load, which is assumed to be known a priori. The problem is optimally solved for a small scenario at the varying of the channel and interface number. It is noted how the use of different ILP models can lead to different relaxations with different optimality property.

Another approach based on a graph model is the one in [40]. Even in this paper the network is modeled with an adjacency graph from which it is derived a conflict graph. The novelty refers to the introduction of a further bipartite resource contention graph where a set of vertices represents the cliques in the contention graph and the other one represents the links. An edge is inserted between each contention region and the set of link belonging to it. Each edge has a maximum capacity equal to the number of available channels in the network, thus accounting for the channels constraint. To account for the radio constraint another set of vertices is introduced representing the nodes in the network. An edge with capacity equal to the number of radios is added between each link and the corresponding couple of nodes. The problem of finding the maximum network throughput is thus reduced to a (modified) max-flow problem in the previously described graph. The solution gives an upper bound for the achievable throughput in the network. Note that no end-to-end traffic is imposed so the optimum solution refers to the optimum traffic pattern for the given topology. The whole problem turns out to be an ILP. In the case of a high number of variables the problem can be relaxed to a LP one. Results are shown in a chain and in random topologies at the varying of the number of channels and node density.

A distributed channel assignment algorithm is presented in [41]. Two performance metrics are defined, one referred to throughput maximization and the other to delay minimization. It is argued that the minimization of the number of interfering edges at each vertex of the conflict graph can approximately achieve both goals. A skeleton assisted channel allocation is proposed. It relies on the construction of a spanning subgraph (called skeleton) of the connectivity graph. The construction is performed by using a distributed algorithm (LMST). The proposed SAFE protocol uses the constructed skeleton to assign channels while preserving connectivity. If the number of radio interfaces is greater than half the number of channels, then a simple random assignment algorithm is used which assures a communication channel exists between any two nodes. If the number of interfaces is less than half the number of channels, then skeleton edges are assigned for connecting neighboring nodes. Nodes which do not share a skeleton edge can use a common channel for their direct communication. Routing issues are solved apart by using a max flow like ILP, enforced with a tunable fairness constraint. The proposed mechanism is tested using graph based simulations and showing the ability in exploiting the multiple channels and radios. Better performance than the joint allocation and routing solution proposed in [42] (described later on) are presented. Simulations at packet level (ns2) show even better improvements, since the ability in balancing the channel occupancy plays an important role in a CSMA scenario where the available aggregated per channel throughput decreases as the number of concurrent NIC increases.

Some distributed approach are in [43, 44].

In [43] a network is considered where all nodes are in the same collision domain, each node can tune more than one interface to a channel, and each channel can be used by concurrent transmission links. The basic hypothesis is that transmissions on the same channel gain a fair share of the available rate. Game theory is exploited to define the allocation conditions under which the selfish choices of each node turns out to be a Nash equilibrium for the system. The goal of each node is to increase its bit rate. It is proved that the stated conditions lead to a Pareto optimal solution, which is also system optimal since it leads to rate maximization. The conditions satisfaction need a central coordinator.

In [44] each node is supposed to be able to listen to all the channels even if it can use only a subset of channels for concurrent transmissions. Available channels are not necessarily orthogonal: a function describing the interference among channels is provided following the approach in [45, 46]. Each node gathers the information about channel occupancy in each channel and then greedily selects the channel which show less interference. The algorithm is proved to converge in a finite time provided some synchronization conditions are satisfied. Moreover the channel assignment can lead to network partitioning. All such issues are fixed by a proposed protocol which is implemented and tested on a real testbed. Some heuristic solutions refer to a 3 way handshake to enforce synchronization, the use of a common communication channel to avoid network partitioning and an interference measure based on two hop information only. The MR-LQSR routing algorithm is then applied.

Routing

Routing is a critical operation in a multi channel network since performance can be substantially increased by correctly loading the links. Imposed link load can dramatically affect the channel allocation decisions. A key concept is expressed in [47] where the channel diversity along paths is considered as a key heuristic to achieve good routing algorithms. The proposed metric accounts for the fact that if different channels are used for the subsequent links in a path, then the spatial reuse can be increased. The proposed metric, called WCETT, is embedded in a DSR-like routing protocol, called MR-LQSR. WCETT depends on the links bandwidth, links average number of retransmission and links channels.

The metric consists of two terms. A first term accounts for the hop diversity, and it is computed as the maximum total transmission delay accumulated by a packet over a particular channel. The minimization of this metric leads to the use of more channels. The second term accounts for the total path delay (regardless of the used channels). The total metric is a linear combination of the two terms. Authors noticed that the first part aims a at a global network optimization by avoiding bottleneck, while the second part aims at a path optimization reducing the e2e delay. The delay on each hop for each channel is computed based on the estimated number of retransmission and the channel rate.

The proposed routing mechanism has been implemented in a testbed, showing good performance.

An enhancement of the WCETT metric is proposed in [48], where the cost in terms of wasted time for the switching among different channels is embedded in the metric. The switching cost indicates if it is worth switching a NIC to a particular channel, based on the estimated fraction of time the card will remain on that channel.

Joint allocation-routing

Problems in this section deal with the case where a traffic specification is provided in terms of e2e traffic demand and the channel allocation problem is jointly solved with the routing one.

A first approach in joint channel allocation and routing is in [49]. Here an iterative algorithm is presented where channel allocation is solved by a greedy heuristic and routing is solved by minimum path or randomized path algorithms. The two parts of the algorithm are solved successively and repeatedly so that the allocation is performed based on the routing outcomes and vice versa. Clearly, at the first step of the iteration an estimation of the link load has to be inferred. This is computed considering all the e2e traffic requirements and a uniform multi-path routing and it is defined as the fraction of paths that traverse the considered link with respect the total number of paths. The iterative process is repeated until no improvement is experienced. Results are compared with the single channel case and with a channel unaware allocation mechanism. In the channel unaware allocation, neighbors of each node are partitioned into groups and to each group it is associated an interface for communication. Partitioning is performed in such a way to balance the number of neighbors on each interface. Some testbed results are also provided.

Clearly, even if interesting, such an approach is not optimal.

The starting point for a mathematical solution can be found in [50] where the problem is described in the single-channel, single-radio case and then extended to the multi-channel multi-radio one. A synchronized TDMA system is assumed. Let consider the single-channel single-radio case. A communication graph and a conflict graph are used to model connectivity and interference. An e2e traffic requirement is specified and a flow optimization problem is defined with the aim of maximizing the throughput offered to that e2e path. The model can account for arbitrary traffic requests and for both single and multiple paths routing. Outcomes of this procedure are the amount of requested load on each potential communication link. No interference issues are considered in this step. A feasible transmission scheduling through time is then found. The traffic requirements determined in the previous point are used to define the fraction of time to be assigned to each link. Interference constraint is also added. Anyway such a scheduling is a NP-Hard problem so that a lower and an upper bound are derived.

The lower bound is derived as follows. It is defined an independent set as a set of nodes which can be scheduled concurrently. It is shown how a convex combination of independent sets is also a valid schedule. An optimization problem can be designed over the polytopes defined by the indicating function of all the independent sets. Such optimization problem aims at maximizing the usage of each node, subject to feasible scheduling and flow requirements coming form the flow problem. This formulation can in principle lead to the optimal solution anyway the polytopes construction requires a non polynomial time. The optimization is thus limited to a subspace of the polytopes, derived from an "easy" collection of independent set. As more set are included in the polytopes, the solution gets closer to the optimum one.

The upper bound is derived using a dual approach. Each clique in the conflict graph indicates that only one of the nodes in the clique can use the channel on a slot. The polytope formed by the indicating function of all cliques an be constructed and an optimization problem can be defined trying to maximize the throughput. This approach does not lead to a tight bound since, in the case of non perfect graphs, some other issues related to the scheduling feasibility should be considered.

A very similar approach is adopted under the physical interference model, where edges in the conflict graph are weighted in order to account for the continuous interference level and an interference threshold is then defined to come back to a binary interference model.

The model easily encompasses the case of multiple channels and multiple radios by adding multiple communication edges in the communication graph. Simple results for the two channel and the two radio cases are shown and compared with the single channel one.

A similar approach is in [2, 51]. Here the network is modeled as a multigraph where vertices represent the nodes and multiple edges are introduced to model both connectivity and interference.

The traffic pattern is specified by e2e requirements between arbitrary nodes. The goal of the optimization is to find the maximum scaling factor which can be applied to the traffic requests and for which a feasible resource allocation can be found. This implies a weighted fairness in the offered throughput to the e2e requests. An optimization problem is constructed defining an indicating variable $y_i + t(e)$ for link e to use channel i at time t. All the constraint related to the number of channels, number of radios per node, number of channels per link and interference free allocation condition are added. The problem is then relaxed letting the variable y to be a continuous variable in the interval $[0, 1]$. It can now be interpreted as a percentage of channel utilization in the following scheduling operation. Clearly the resulting LP problem gives only an upper bound of the optimal solution, since the found solution could be not schedulable. The problem is completed by defining a feasible channel assignment and transmission scheduling. This is an NP-Hard operation so two main heuristic based on set coloring and maximum packing are proposed. Results at the varying of number of channels and number of NICs are shown in different network topologies.

A similar optimization problem is in [52] where a different solution procedure is proposed. Traffic is supposed to be generated by multiple sources and directed toward a single gateway node. Even in this case a flow problem is defined as the problem of maximizing the scaling factor for the traffic demands subject to a feasible scheduling. A second LP problem is thus defined, where the scaling factor is fixed and which aim is to minimize a metric related to the interference level. Previous optimization does not account for the number of card constraint. This is fixed by an heuristic algorithm which modifies the flow assignment and determines a new scaling factor so that a feasible NICs and channel assignment is possible. A further step defines a new LP problem which aim is to redistribute the flows such that none of the previously fixed constraint are violated and the interference level is reduced. Then a new flow scaling assure that all the interference constraints are satisfied. Finally a scheduling algorithm is defined. The whole algorithm is proved to attain a constant factor approximation of the optimum solution. During the solution process a lower bound to the network throughput is also derived.

All the proposed models assume perfectly orthogonal channels.

In [45, 46] it is pointed out how the use of partially overlapping channels can improve the performance of a network by allowing an higher spatial reuse. Some experimental results report the interference between concurrent transmissions as a function of the channel overlap and devices distance. This results show that partially overlapping channels can be considered orthogonal if the transmitter and the receiver are sufficiently far apart. A model for the overlapping channels interference is presented which defines a metric based on the overlapping area between the channels spectrum masks provided by the standard. Such a new measure is introduced in the channel allocation mechanism originally described in [53] by enhancing the interference evaluation mechanism and, allowing the use of more channels, shows a performance improvement. The notion of interference between partially overlapping channels is then formalized in such a way it can be embedded in many of the proposed multi-channel multi-radio allocation algorithms. In particular it is introduced in the analysis provided in [52] where an upper bound for the throughput is derived. A great throughput improvement is shown with respect to the case when only orthogonal channels are used.

2.3.3 Protocols

In the following, some practical oriented solutions are presented, which are able to exploit multiple channels in ad hoc networks. These mechanisms are based on heuristic considerations and are designed accounting for practical issues for their implementation.

Single radio

The protocol proposed in [54] (SSCH) considers the case of an ad hoc network, where multiple channels are available, but each node is provided with a single channel half duplex NIC. In particular the paper considers the use of legacy IEEE 802.11a adding a standard compliant protocol which allows for the use of all the available channels. The protocol is completely distributed and only a small amount of control traffic is needed. It works in multihop environments, and can exploit spatial reuse.

The time has to be considered as divided in subsequent fixed length time slots. Authors claim that synchronization is not a hard requirement. The main idea is to provide a node with a channel hopping sequence indicating to each NIC the working channel on each time slot. The hopping sequence is computed using a modulo operation based on a seed value and a starting channel index. Synchronization among nodes is obtained by exchanging the seed and the channel index among neighboring nodes. Each node synchronizes with the hopping sequence of the node to which it has traffic to send, i.e. each node can change its hopping sequence based on its traffic requirements. The proposed MAC is shown to obtain a higher throughput than legacy 802.11a. Anyway a higher delay (an jitter) is shown, which impact on the TCP performance.

A different use of channels hopping is considered in [55]. All the nodes have to be synchronized with the hopping sequence. At each frequency hop, a time interval is reserved for a transmitter initiated handshakes between transmitters and receivers. The winners of such handshake stop their frequency hopping and start to transmit data on the current frequency, while all the other nodes continue to hop. An analytical model for the throughput is provided and the results are compared with the ones of a multichannel ALOHA with receiver oriented channel assignment.

The protocol RICH-DP in [56] works in the same way but the handshake is initiated by the receiver, which uses polling among neighboring nodes.

The protocol proposed in [57] considers a classic CSMA MAC, but a control channel is defined which is dedicated to RTS/CTS control traffic. The other available channels are devoted to data traffic. In particular, once fixed a total bandwidth W, a portion W_c of it is used for control traffic and the remaining is subdivided in M data channels. To start a communication a node sends a RTS on the control channel. RTS contains the list of free channel as perceived by transmitter. The receiver, based also on its own free channel list, responds with a CTS containing the channel to be used (it is assumed communication uses only one channel). The system is simulated as a function of W_c and M and it is shown that this MAC outperform standard 802.11 both from the throughput and delay point of view.

Another approach (MMAC) using a dedicated control channel is in [58]. Here each node is provided with one 802.11 interface. A common channel is used to exchange control information and to decide the channel to be used for data communication. The control channel is realized reserving periodic time intervals on a commonly selected channel. Synchronization is obtained through a beaconing system similar to the one used in the power save mode of standard 802.11. The agreement on channel usage is obtained using an RTS/CTS handshake where simple channel usage metric is exchanged. After the selection, cards of transmitting and receiving nodes are set to the chosen channel. The proposed mechanism allows for eliminating the multichannel hidden node problem, since the information on control channel are heard by all nodes around both transmitter and receiver. It is shown that the proposed solution reaches an improvement up to 300% by using 3 channels in the all-in range case and up to 200% in the multi-hop case.

The paper [59] considers the case of a relay network applied to a cellular system. A protocol for the relay network formation is presented which creates a tree-like topology toward the gateway. Relay nodes are provided with a single interface but can use multiple channels. Based on the previously formed topology, channels are assigned along paths in order to reduce interference and increase spatial reuse. A simple distributed algorithm is presented where each node collects the channel usage in the neighborhood, by using control messages, and selects the least used channels. The selection mechanism is compared to the optimal solution provided by [52] showing the ability to reach 80-85% of its performance in this scenario.

Multi-radio

The Multi radio Unification protocol [60] considers a legacy 802.11 network where each node is provided with multiple cards, each working on a fixed channel. The protocol works over the LL layer so it can be realized over the driver level. Each node holds a neighboring table where it is stored the address of all neighboring nodes cards, together with the channel quality indicator. Such an indicator is calculated based on the RTT provided by probe packets. Communication between nodes take place by using the best channel. In order to balance the use of channels, some strategies for sharing of the traffic over multiple interfaces are considered, but none of them have shown improvements over the MUP.

In [48] the number of per-node 802.11 cards is less than the number of available channels. The proposal refers to a link layer protocol (HMCP) for the use of multiple cards and a routing protocol to exploit the particular scenario. The authors propose to provide each node with a set of interfaces fixed on some particular channels, and another set of switchable interfaces which can be dynamically tuned on different channels. Fixed interfaces channels are set using two hops information exchanged by using “hello” packets. The goal of such an assignment is to balance the use of channels dedicated to fixed interfaces in a two hop range. Fixed interfaces are used for reception: each node that has traffic to transmit has to switch to the fixed channels of it destination node. To each channel it is associated a separate queue: switchable interfaces are switched to the longest queue. To enforce the fairness a timeout is also used to force the switching operation.

A centralized channel assignment algorithm and a related protocol for its implementation is proposed in [61]. To avoid topology modification a common channel is assumed to be used by all the nodes. The common channel is selected by a centralized algorithm which aims at minimizing the interference. The remaining channels are allocated as follows. Each node is supposed to gain information on the potential interfering neighbors by sniffing the traffic in the channel. This implies that only nodes on the transmission range are considered, since nodes which packet can not be correctly received are not considered. Interference information are exchanged in the two hop scope. Based on the collected information a new conflict graph is constructed which directly accounts for multi radios capability at each node. In the conflict graph there is a vertex corresponding to each radio. A list coloring problem is solved by means of an heuristic algorithm (BFS-CA) which selects channels giving more priority to the ones near to the gateway. Simulations under different scenarios are shown and the proposed algorithm is compared with a distributed greedy approach where each node set its radios on the least interfered channels. Throughput results shows that in some particular scenarios the greedy approach shows better results than the BFS-CA, but in most cases BFS-CA improve the network performance. The algorithm is tested also on a real testbed.

A distributed protocol for channel assignment is in [42]. The network is supposed to have gateways that connect the mesh nodes to other networks. A distributed route construction which creates a tree between nodes and gateways is presented. NICs in a node are classified as UP NICs, for the connection towards the gateway, and DOWN NICs for the connection toward leaves. Channels are associated to the NICs based on a occupancy measure which is calculated using periodic information exchanges about channel load. Such information is collected and used to sort the channels according to their total occupancy level. The least occupied channel is chosen first. An extensive simulative results set is provided, showing the effect of parameters changing. In particular the proposed protocol is proved to perform very close to the centralized one previously presented in [49]. Results on an actual testbed are also presented.

2.4 Rate adaptation in IEEE 802.11 WLAN

A number of recent studies deal with the problem of 802.11 rate adaptation (RA). The majority of proposals aim at optimizing the PHY rate with respect to channel impairment only, with a few exceptions.

Some RA algorithms makes use of the Received Signal Strength Indicator (RSSI) to select the PHY mode. The RSSI, in fact, is a measure of the received signal power, which shall be proportional to the Signal to Noise Ratio (SNR) at the receiver. In practice, however, this approach is limited by a number of factors. For instance, it is a common experience that the RSSI returned by a wireless board circuitry is not always reliable. Furthermore, some schemes select the PHY mode according to the RSSI measured at the transmitter, assuming it is the same that would be experienced at the receiver. However, the assumption of symmetry is often disattended in reality.

In [62], the authors propose the MPDU-Based Link Adaptation Scheme (MBLAS), which makes use of an analytical model to evaluate the 802.11 goodput as a function of the SNR, the PHY mode and the payload size. The proposed model takes into account the 802.11 backoff and retransmission procedure, but it is limited to the scenario with a single transmitter/receiver pair, for which MBLAS provides the theoretically optimal rate. However, the scheme is suboptimal in multiple stations scenarios.

In [63] the authors propose an RSSI–based Link Adaptation strategy. The PHY mode is selected based on the measured RSSI, which is compared with dynamically defined thresholds. The use of dynamic thresholds aims at alleviating both the inaccuracy of RSSI measurements and the channel asymmetry issues. The drawback of this proposal is that the thresholds are adjusted considering the loss rate observed for a given PHY mode: this practice could easily lead, in case of frame losses due to collisions, to an undesired decrease of the thresholds.

In the Receiver Based Auto Rate (RBAR) [64], the receiver STA selects the most suitable PHY mode on the basis of the RSSI measured during the reception of RTS frame. The selected PHY mode is, then, communicated to the sender by using the CTS frame, so that the sender will adopt the chosen rate for the subsequent data transmission. While effective in overcoming channel asymmetry issues, it is to be noted that this algorithm is not standard compliant since it requires modification to the RTS, CTS and data frame structure, as well as to the PLCP header, in order to include the necessary control information. Moreover, the proposed RSSI-based rate selection scheme at the receiver takes into account only the success probability of a single frame transmission, thus completely neglecting the impact of the MAC layer on the performance.

Another well known RA algorithm is the Auto Rate Fallback (ARF) [65] which is based on the following consideration. In the absence of interference from other users, a certain number of subsequent failures are likely due to a lowering of the SNR, so that a more robust rate has to be selected. Conversely, when a certain number of subsequent successful transmissions is observed, a higher rate is selected to improve throughput. This type of schemes is not subject to RSSI measurement inaccuracy nor to channel asymmetry issues. One of the drawbacks of ARF, however, is that it periodically tries a higher transmission rate[1] to check if it is sustainable; this behavior is inefficient in static scenarios where the optimal rate remains the same for prolonged periods. The Adaptive Auto Rate Fallback (AARF) [66] aims at alleviating this problem by applying a binary exponential backoff to the number of subsequent successful transmissions needed to try a higher rate. In this way, AARF is more stable than ARF and achieves better performance in static scenarios. Nonetheless, both ARF and AARF assume that packet losses are always due to channel errors, so that their performance can rapidly degrade in high traffic scenarios, where a significant amount of packet losses are caused by collisions.

Some other RA schemes try to combine the best features of the RSSI-based and loss-based approaches. For instance, the Hybrid Rate Control (HRC) [67] exploits the measured RSSI and Frame Error Rate to distinguish between short-term and long-term variations of the channel conditions. This mechanism exploits a throughput-based rate controller which probes adjacent rates to determine if a rate switch is necessary. Moreover, two sets of thresholds (named stable and volatile low thresholds) are used depending on the detected variations of the RSSI. Again, this scheme does not consider the fact that packet losses might be also due to collisions.

To summarize, a major drawback of all the RA schemes discussed so far is that they are designed for scenarios in which a single node is transmitting on the wireless channel. In real situations, however, it is often the case that multiple nodes contend for the medium. Consequently, due to the way the 802.11 MAC works (i.e., CSMA/CA with DCF), the goodput actually experienced by active nodes is influenced not only by channel-related packet losses, but also by MAC collisions and variations in the time required to access the medium. These factors cause the formerly discussed RA algorithms to achieve sub-optimal and, in some cases, very low performance. In particular, loss-based RA schemes such as ARF or AARF do not work properly, since losses due to MAC collisions can easily lead to the choice of a low-rate PHY mode even in cases in which a high rate is sustainable. As for RSSI-based schemes, it is to be noted that the choice of RSSI thresholds is optimal only for the single user scenarios, but can easily become non-optimal as the time required for

[1] As reported in [64], ARF tries a higher rate every 10 successful transmissions in a row.

a successful packet transmission increases due to collisions and increased medium access delay.

In more recent years, some solutions have been proposed to address this problem. For example, Closed Loop Adaptive Rate Allocation (CLARA) [68] is an ARF-like RA scheme which aims at reacting differently to losses due to channel errors and collisions, respectively. A significant drawback of this scheme is that it is based on the assumption that losses after a successful RTS/CTS exchange are always due to channel errors; consequently, it requires the use of the RTS/CTS handshake that has a significant cost in terms of overhead. The Collision–Aware Rate Adaptation scheme proposed in [69] exploits the same mechanism for loss differentiation but implements an adaptive RTS/CTS probing scheme which reduces the overall RTS/CTS usage, thus somehow mitigating the inefficiency of CLARA. We note, however, that both CLARA and CARA do not consider the impact on the performance of the variations in the medium access time.

To conclude, no previous work, to the best of our knowledge, has provided a RA scheme which is optimal with respect to both channel impairment and contention-related issues, comprehensive of both frame collision probability and medium access times. In particular, no analytical models for goodput performance in multi-user scenarios have been presented. In the next section we propose such a model, which enables the definition of our Goodput-Optimal Rate Adaptation (GORA).

2.4.1 Application layer perspective: VoIP model

Rate adaptation techniques considered in the previous section might not be specifically suited for optimizing the performance of some kinds of applications. In this section it is introduced a model for the quality evaluation of VoIP communications, which links the perceptual quality to some basic traffic metrics such a as throughput, delay and jitter. Such a model will be used to define an optimization algorithm for enhancing the performance of VoIP links.

Playout buffer and quality evaluation

The voice codec considered in this section produces CBR traffic flows, anyway the transmission link introduces a random delay such that the received packet delay presents a jitter which can decrease the voice quality.

To reduce this problem, a common solution refers to the introduction of a buffering mechanism at the receiver which is used to add a delay to the early packets in order to reduce the jitter. Thus, jitter is traded for an increased average delay.

The playout buffer is generally arranged to operate as a first-in/first-out (FIFO) buffer in which voice packets are placed when they arrive at the decoder. Then, after an initial delay, voice packets are fetched from the playout buffer at the same (constant) rate they were created by the coder. Packet sequence numbers can be used to ensure the data is played out in the correct order and to detect packet loss.

A variety of techniques have been proposed for the management of playout buffers. Typically, these techniques involve estimating the maximum variation in the transmission delay expected for each packet as it passes through the network. The size of the playout buffer is chosen such that variations in packet arrival time of this order can be smoothed. The end-to-end delay introduced using this approach is then approximately equal to the maximum jitter estimated for the packet data stream. Packet losses may occur in two cases, namely when the playout buffer is either full when a packet arrives, which is referred to as an overrun, or empty when a packet is due to be played out, which is referred to as an underrun. Overruns can be avoided by properly dimensioning the buffer size, while underruns can be reduced by increasing the playout delay (and, hence, the end–to–end delay).

In [70] a revised version of the E–model is presented as the performance metric for VoIP communications [71, 72]. The quality evaluation function produces a rating R of the voice quality in a scale from 1 to 100 (70 corresponds to the PSTN quality), as a function of the system state (characterized by P_{loss} and m_s). A translation to the Mean Opinion Score (MOS) is provided by the following relation:

$$MOS = \begin{cases} 1 & R < 0, \\ 1 + 0.035R + 7 \times 10^{-6} R(R-60)(100-R) & 0 \leq R \leq 100, \\ 4.5 & R > 100. \end{cases} \tag{2.11}$$

The relation in (2.11) is represented in Figure 2.5, while Tab. 2.4.1 gives a classification of the voice quality with respect to the MOS and R-factor values.

R-factor	**Quality of voice rating**	**MOS**
$90 < R \leq 100$	Best	4.34 - 4.5
$80 < R \leq 90$	High	4.03 - 4.34
$70 < R \leq 80$	Medium	3.60 - 4.03
$60 < R \leq 70$	Low	3.10 - 3.60
$50 < R \leq 60$	Poor	2.58 - 3.10

Table 2.2: R-factors, quality ratings and the associated MOS

The metric is composed by many terms accounting for codec type, packet loss, end to end delay. In [70] the following metric is proposed:

$$R = 94.2 - I_{el} - I_{ec} - I_d \, . \tag{2.12}$$

In (2.12), I_{el} depends on packet loss rate and a graphical interpolation for different codecs is provided in Fig. 2.6. An approximated expression is provided below

$$I_{el} = 34.3 \ln(1 + 12.8\, P_{loss})$$

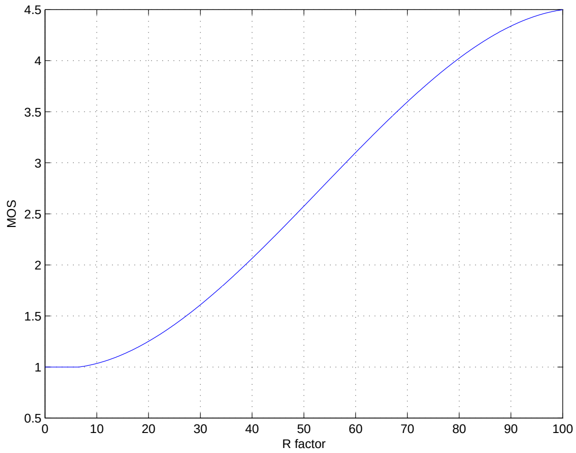

Figure 2.5: Relation between the E-model rating R and MOS rating

where P_{loss} is the e2e packet loss probability accounting for packet losses due to the link and to the buffer.

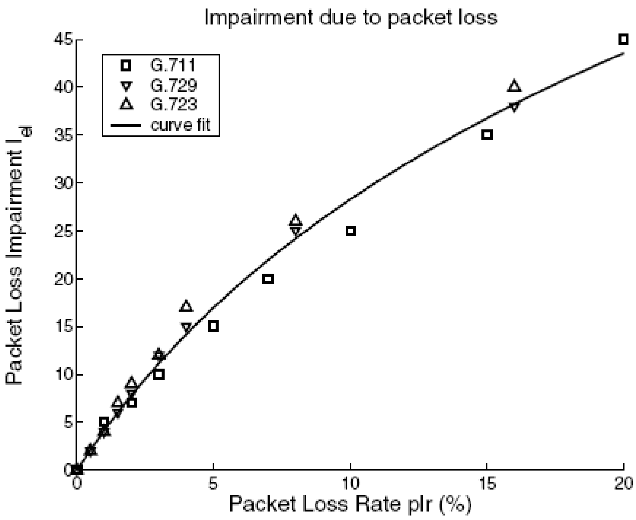

Figure 2.6: Pkt loss dependency

The term I_{ec} accounts for the impairments due to the voice compression performed by the codec. A graphical representation of such a factor is given in Fig. 2.7. Unfortunately, there exists no simple mathematical expression for such a term.

Finally, the term I_d is the delay impairment factor, which depends on the mouth–to–ear delay d. An approximation of I_{di} is provided below (the exact formula is in ITU recommendation):

$$I_{di} = 24\text{d} + 110(\text{d} - 177.3 \cdot 10^{-3})H(\text{d} > 177.3 \cdot 10^{-3}) \ ; \tag{2.13}$$

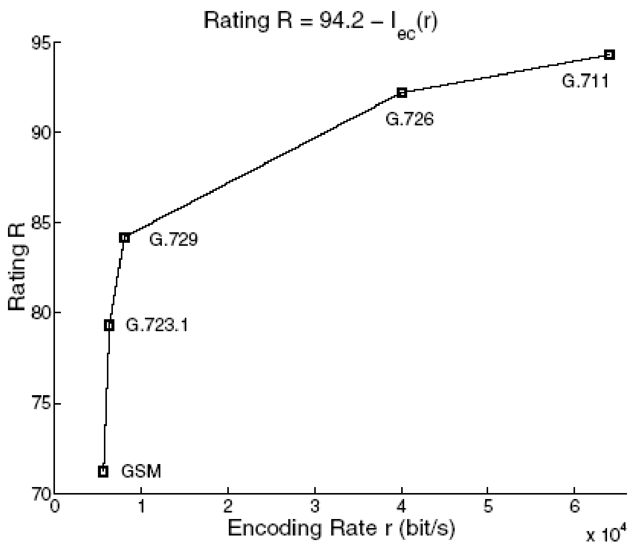

Figure 2.7: Rate dependency

where $H(\cdot)$ is the Heavyside function (unitary step function). The relation in (2.13) is represented in Figure 2.8.

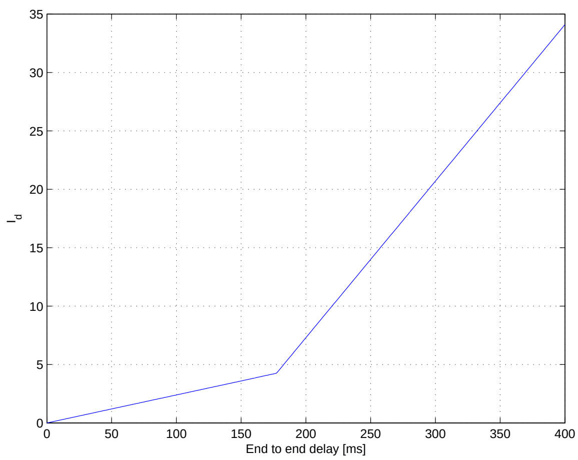

Figure 2.8: Delay impairment factor as a function of the one-way delay

The mouth–to–ear delay, d, is inclusive of the algorithmic and packetization delay associated with the codec and the IP packet processor, m_c, the one way network delay $m_{s'}$, and the playout buffer delay δ_{buff}, so that we have

$$\mathrm{d} = m_{s'} + m_c + \delta_{buff} \ .$$

The term m_c depends on the specific codec used and the number N of voice frames aggregated in the same IP datagram. In case of G.711, we have

$$m_c(G.711) = N \times \tau_v \ ;$$

with $\tau_v = 10\,ms$. G.729a codecs, instead, use a $\tau_w = 5\,ms$ look ahead in order to encode the current $10\,ms$ PCM encoded block, so the coding delay becomes

$$m_c(G.711) = N \times \tau_v + \tau_w \ .$$

This metric relates the perceptual quality to some standard traffic related metrics (delay, jitter, throughput) and allows for applying an optimization algorithm which chooses the transmission strategy in order to improve the quality of the VoIP connection, as will be described in Section 4.6.

2.5 Scheduling in FDMA cellular networks

Orthogonal Frequency Division Multiplexing (OFDM) is the most widespread and promising solution for multiple access and signaling in today's wireless (broadband) networks. Its deployments include WLAN physical layer implementations such as IEEE 802.11a/g, ETSI HIPERLAN/2, the IEEE 802.16 standard for broadband wireless access in metropolitan area networks and the Digital Audio/Video Broadcasting (DAB/DVB) standards. The OFDM technology is based on the principle of multi-carrier transmission, originally appeared in the design of high speed digital subscriber line (HDSL) [73]. The OFDM transmission method results to be a really effective platform in multi-path environments with frequency selective fading. A significant advantage of the OFDM technology is the possibility of allocating power and rate optimally, by using adaptive modulation according to instantaneous subcarrier quality, thus maintaining acceptable BER per subcarrier [74]. Moreover, in the multi-user scenario, it is possible to assign subcarriers to the less interfered user, owing to the channel diversity among users placed in different locations.

Different users can in fact experience different channel conditions at a given time instant and frequency band and this property can be exploited by opportunistically allocating the resources to a subset of users which have the best conditions. This property is usually referred to as *multiuser diversity* [75].

Depending on the channel status, each user can experience different channel attenuation on each subcarrier and at different time instants. This property is usually referred to as *frequency and time diversity*. This degree of freedom can be exploited for increasing the transmission performance by appropriately allocating the resources in frequency and time.

Generally speaking, the optimization problems arising in such systems would require to jointly optimize

- users selection / flow scheduling

- subcarrier allocation to the users
- bit loading on each subcarrier
- power loading.

Different optimization goals can be pursued

- flow level: fairness / minimum per-flow rate / maximum throughput
- physical level: power minimization / maximum power constraint

Clearly, all the optimization variables and also different optimization goals are strictly interrelated such that a joint optimization is required for achieving optimum performance.

It is worth noting that variables indicating the allocation of a subcarrier to a user are usually considered as boolean variables in order to model the constraint that a subcarrier can be allocated to a single user at a given time. In this case, the joint optimization problem, accounting also for the bit loading and the power selection, becomes a mixed integer problem, which complexity can represent an issue for practical implementations.

Moreover, different scenarios can also be considered. Most of the work in the literature deals with single cell systems, thus neglecting the inter-cell interference. Multi-cell systems, where concurrent transmissions are allowed from different neighboring base stations, represent a much challenging (and realistic) environment, since the inter-cell interference couples the optimization problem along all the cells and in general would require a network wide solution. Consider as an example the case of the throughput maximization in an isolated cell. The allocation of subcarriers with a high bitload is the most desirable solution since this allows for an higher throughput and also an increased number of admitted users, which can also be beneficial for increasing the fairness among flows. The same objective in a multicellurar environment turns out to be much more challenging as the use of high bit loads requires high transmission power, which in turn causes high interference to neighboring cells reducing the spatial reuse. In this case a trade off between modulation efficiency and channel reuse has to be considered.

Many proposals addressing this kind of optimization problems and spanning a subset of the previously listed optimization scenarios, variables and goals, have appeared in the literature.

2.5.1 Allocation

In this section it is specifically reviewed the work related to the subcarriers allocation, i.e. the problem of assigning each physical resource to traffic flows (allocation), by also deciding the more appropriate modulation (bit loading) and the correct power (power loading).

In [76] a joint allocation, bit and power loading algorithm has been presented. Here the problem of channel allocation and bit loading in an isolated cell is subject to a hard minimum rate requirement from each allocated flow. The objective is represented by the minimization of the transmitted power. The allocation problem, which is a combinatorial integer problem, is relaxed by allowing a fictitious time sharing of the available subcarriers, thus obtaining a lower bound for the transmitted power. The solution is brought back to the integer domain by means of an heuristic mechanism, allocating each subcarrier to the user with the highest time sharing. The outcome of such an allocation is then used to run a single user water-pouring algorithm for the actual bit loading.

The work in [77] describes a subcarrier allocation and power adaptation in a single cell, with the additional aim of providing fairness among allocated users. Bitload is considered fixed and equal for each of the available subcarriers. A modified version of the ideal GPS scheduler is used to define the number of subchannels to be allocated to each user in order to achieve a fair resource allocation. Thus, subcarriers are allocated to users with the aim of minimizing the transmitted power. If the power is below a given threshold, the allocation is considered successful, otherwise the number of available subcarriers is decreased by 1 and the procedure is repeated.

Channel allocation together with power adaptation techniques for throughput maximization are presented in [78], in the case of a generic multiple access schemes with orthogonal channels. The problem is addressed in a multicellular scenario, thus accounting for the inter-cell interference. Three centralized heuristics are developed. Heuristics are based on the use of efficiency metric computed starting from the useful and interfering channels gain. In the simpler one, for each subcarrier the user with the best channel is allocated. The same subcarrier can be reused by a concurrent allocation from a neighboring base station, only if the added transmission leads to a global throughput increase. It is shown how starting from this heuristic, it is more efficient to adapt the modulation rather than the power in order to improve the network throughput. The joint solution is only slightly better than each of them. No fairness issues are addressed.

In [79] a OFDM downlink in a single cell system is considered. To each user it is associated a utility function: the design of such functions (or the design of the marginal utility functions) allows for differentiating the service provided to the users.

The problem of optimal joint DSA (dynamic subcarrier allocation) and APA (adaptive power allocation) is considered with the aim of maximizing the utility of the system. System utility is the sum of users' utilities. If the utility is proportional to the received rate, the problem turns out to be the throughput maximization. DSA and APA are solved separately, and then a joint solution is shown to obtain better throughput. Some interesting consideration on the design of utility function are made, explaining the existence of a global optimum and proposing utility functions which aim at obtaining maximum throughput, weighted fairness or max–min fairness. In [80] practical algorithms for the solution of the DSA and APA optimization problems are proposed.

In [81] two cases are considered: a rate based utility, accounting only for channel status and a delay based strategy accounting also for queue status. It is argued that only queue aware scheduling can achieve the maximum stability region (MSR). MSR policies do not account for QoS differentiation but this could be fixed by appropriately designing the utility functions. It is shown how the use of an exponential utility function brings to the well known proportional fairness scheduling algorithm. It is also stated that the class of utility function with polynomial derivative are able to achieve the "maximum stability region".

2.5.2 Scheduling

Link Layer scheduling algorithms for packet switched networks have the goal of achieving a fair allocation of the bandwidth resources to the flows competing for the access to the shared medium. The basic packet scheduling schemes have originally been proposed for wireline networks, where the channel is usually assumed to be error-free and of constant capacity [82].

Known packet scheduling schemes have been extended to wireless networks, by taking into account the additional feature of a strongly time-varying channel [83] and addressing the problem of power efficiency.

If fairness constraints were not taken into account, mere throughput maximization would have an extremely unfair outcome, where few users (those enjoying good channel conditions) are repeatedly allocated most of the bandwidth, while the others starve.

On the other hand, efficiency at the physical layer has to be pursued by exploiting the multiuser diversity "riding the peak"of the channel gain variations [75] and thus allocating resources to the user experiencing the best channels conditions.

In some scenarios it is unrealistic to pursue short-term fairness, due to the specific characteristics of the radio medium. In fact, if the scheduler aimed at achieving fairness among flows in the short term, the performance of the system would be far from optimal, as we would schedule users experiencing a bad channel state, without any benefits for their own flow nor for the aggregate network throughput. It has long been recognized that a better policy is to allow for some short-term unfairness in order to improve efficiency.

Clearly, a trade off between power efficiency and fairness arises.

According to that requirements, basic schedulers should keep track of how much data each flow transmitted in the past, and compensate lagging flows when their channel conditions improve, or when they have been starving for too long. To implement this mechanism, the scheduler needs to take into account both channel and traffic state information.

A formalization of that mechanism is in [84, 85], where a unified framework for multiuser single channel scheduling is presented. A Gilbert-Elliot channel model is considered as a reference model and a equal physical transmission rate for all the users is assumed.

Following the proposed approach, every scheduling algorithm is composed by the following functional blocks:

- error free service (e.g. WRR, STFQ, WFQ, WF2Q): it defines the service that would be obtained by the traffic fluxes in the idealized case of no transmission error. The basic reference mechanism is the General Processor Sharing scheduling. Such algorithms are derived from the wired networks.

- lead and lag model: specifies how the service provided to a flux is measured.
- compensation model: it acts in a dynamic way to compensate the received service by different users, in order to approximate a fairness model such as the ideal GPS scheduling.
- slot queues and pkt queues: lagging a leading metrics can be based on transmitted packets or experienced delay. This is used to decouple delay and throughput measures.
- channel monitoring and prediction: usually a simple on–off model is assumed and the status of the channel is known at the moment of scheduling decision.

Different instantiation of such blocks lead to different scheduling algorithms: CSDPS, WPS, IWFQ, CBQ-CSDPS.

In particular in [85] some analytical bounds are presented comparing the performance of different scheduling mechanisms accounting for short and long term fairness for backlogged flows, delay-throughput achievable region, short term bounds for users with clean channel, long term delay and throughput bounds for users with error bounded channels.

It turns out that the most valuable algorithms are CIF-Q [86] and WFS [87].

As already pointed out, channel status information is required for optimizing the power allocation and increasing the throughput. In case of a stochastic channel evolution, a prediction of the channel status in the near future is essential for achieving good performance. In [88], starting from the framework in [85], it is proposed a scheduling algorithm similar to WFS, which accounts for a multi-state Markov channel. Each pkt can be retransmitted if in error, but it has a finite Time To Live: using the channel prediction, based on the Markov model, the user which has the highest probability of correctly receive a pkt is scheduled. An adaptive FEC is also used. It is shown that this algorithm performs better than WFS in terms of throughput and delay, preserving the fairness properties.

Some other consideration need be made when trying to implement such algorithms.

Many packet scheduling algorithms proposed in the literature rely on the time provided by a common reference clock in order to implement the error free model and the compensation blocks; packets are tagged as they enter the queues according to the clock time. The selection of the next packet to transmit takes place by taking the time tag and the packet length (thus, the packet transmission time) into account [89].

Virtual time-based algorithms, however, have a high computational complexity, which makes their implementation difficult and expensive. Thus, algorithms based on credits, such as CBFQ [90], have been proposed. They are simple and computationally efficient, achieve fairness among flows and can be adapted to work in a wireless environment (see, e.g., WFCQ [91]). In credit-based algorithms, the service state of each flow is summed up by a single number, its credit value. A flow gains credits when it is not scheduled, and uses credits when it is scheduled. This scheme makes it easier to take a continuous channel model into account; e.g., in WCFQ channel quality is dealt with by introducing a cost that depends on the state of the channel: the scheduling priority of a flow will depend both on the amount of credits it has accumulated and on its channel quality. A flow experiencing a bad channel is at first prevented from transmitting; it is scheduled again when either its channel quality has improved or it has accumulated enough credits so as to overcome its bad channel quality index.

A more theoretical approach is in [92], where the optimal strategies for achieving (i) maximum throughput subjected to temporal fairness constraints or (ii) general utility function maximization or (iii) minimum performance guarantees are presented. The optimization is intended in the asymptotic sense. In the case of non stationary condition a stochastic approximation algorithm for the solution of optimization problem is presented. Here the channel model is not specified since its evaluation is included in the utility function.

Alternative formulations of the scheduling problem are also possible.

An approach based on Learning Automata is presented in [93]. Here ACK and NACK are used to estimate the link status for each user. To the users it is associated a probability to be chosen in a scheduling round. The probability is calculated based on a fairness function and on channel feedback. The transmission rate is selected based on a probability distribution calculated using past history. The algorithm converges in stationary situations and dynamic scenarios are also tested.

Previous works never account for traffic sources dynamic and queue stability (since an access level perspective was considered and a heavy load condition was assumed). Borst in [94] considers the case of scheduling at each time slot the user with the best channel with respect to the its mean channel status. The obtained service is then used to evaluate the stability region of the system. The study is performed in an analytic way. It is shown how the Proportional Fairness scheduling algorithm falls within the analyzed framework.

Tassiulas proposes an analytical study of the scheduling in such scenario. In [95] it is assumed a ON-OFF channel model for each user and arbitrary traffic arrival pattern. It is proved that a simple rule based on queue length can attain the optimum weighted throughput. No QoS issues are considered, since a symmetric channel status for all users is assumed and the aim of the optimization is not the stability of all queues.

Shakkottay [96] provides an analytical characterization of the “exponential rule”, that is a particular scheduling rule which is able to obtain optimum throughput exploiting information on channel and queue length. The channel is assumed to be modeled by a Markov process with multiple possible transmission rates. No QoS issues are considered, anyway the stability of the system implies the stability of each queue.

The work in [97] considers a framed TDMA. Slot in a frame are assigned in order to guarantee the QoS requirements in terms of delay and throughput to QoS users, while optimizing the network performance. The number of slots assigned to a user is computed in order to guarantee the target delay and pkt loss, given the channel model. An admission control is assumed for QoS users. Remaining slots are assigned to best effort traffic using WFS.

Packet schedulers described in the literature have traditionally been designed for TDMA systems with a fixed physical transmission rate, where the goal of the scheduler is only to select one flow at a time for transmission. Thus, such solutions did not tackle the problem of simultaneously scheduling packets belonging to different flows and allocating them a pool of transmission resources with possible diffrent physical rates. In a way similar to the one proposed in [85], the work in [98] propose a way to schedule users over multiple channel decoupling the problems of throughput optimization and fairness. A first block performs the weighted allocation of the available channels to the users solving a LPI problem, while a second block performs the weights update, using a stochastic approximation method. Information on available channel rate is exploited.

2.6 Motivation for the proposed contributions

The literature review proposed in this chapter shows how performance optimization in wireless networks represents an open research field where many issues remain unsolved. The need for improving the performance by simultaneously acting on many parameters is such that the concept of cross-layer design is especially suited for these kinds of networks.

Anyway, a comprehensive and solid analytical model is lacking. Nonetheless, a tentative model has been recently presented which is denoted as "layering as optimization decomposition" and allows for a systematic definition of the different layers functions. The basic idea has been described in this section.

In Chapter 3 this model will be used for optimizing the performance of a multi-channel multi-radio ad hoc network, by defining a joint algorithm for channel allocation, congestion control and scheduling. These kinds of networks inherit all the optimization issues of classic ad hoc networks, with additional degrees of freedom in the optimization process due to the availability of multiple channels and multiple interfaces. Asymptotic results for the achievable capacity as a function of the number of channels and interfaces have been proposed which shows the potential performance. Some analytical approaches which uses classic optimization theory and complex heuristics to determine more realistic bounds have been also described. Such algorithms do not clearly show the cross-layer aspects and are far from being implemented in practice. Moreover, practical mechanisms for exploiting the presence of multiple channels and interfaces have been described which are essentially based on heuristic argumentations. Thus, practical algorithms based on solid analytical background are lacking. The proposal of a framework for applying cross-layer algorithms based on analytical argumentations in the context of multi-channel multi-radio ad hoc networks is one of the contributions of this thesis.

The presence of multiple channels in cellular network is a more classic aspect. Nonetheless, efficient algorithms for allocating the available resources to the mobile users are being investigated. Some of these studies related to scheduling and allocation of the physical resources have been presented in this section. In this context there are only few proposal regarding scheduling algorithms in FDMA networks, whereas most of the contributions are focused on efficient allocation at the physical layer, but are mostly tested in single cell scenarios. Merging the two aspects requires a cross-layer investigation, which have been approached from a theoretical point of view by some authors. In this context, simple algorithms which preserve the modularity of the layered structure, but at the same time allow for an interaction between the scheduling and resource allocation mechanisms, represent an interesting challenge. In Chapter 5 a framework for achieving such a goal will be presented.

Moving toward more practical issues, the problem of optimizing the performance of a standard IEEE 802.11 network is another field open for research contributions. In particular, the problem of adapting the transmission rate to the link conditions by complying with the standard specifications has not been fully investigated yet. Theoretical and practical solutions have been proposed which aim at optimizing the link throughput by means of heuristic algorithms. Optimal solutions, which are not standard compliant, have also been proposed for the case of absence of interfering transmissions. Thus, an optimal solution for the case where the medium is shared with concurrent transmissions has not been presented yet. Moreover practical and efficient algorithms for the same scenario are lacking. One of the contributions of this thesis is the analytical definition of the optimal rate adaptation for an IEEE 802.11 link in the presence of concurrent interfering transmissions. Practical issues will also be addressed by allowing the definition of an algorithm which could potentially be implemented in actual network cards. The proposed general framework allows to consider different objectives for the rate adaptation and to jointly optimize the rate with some other tunable parameters, such as the maximum retransmission limit.

Chapter 3

Optimization in Multi-Channel Ad Hoc Networks

With the motivation of improving the performance of multi-hop wireless networks, in the last few years great attention has been devoted to networks where each node is provided with multiple radio interfaces and can operate on multiple channels. This new degree of freedom has been proved to potentially allow for achieving the full capacity even with a reduced number of interfaces per node [36].

In this chapter, we consider the problem of joint congestion control, channel allocation and scheduling for multi-hop multi-channels wireless networks in a general communication and interference scenario. The problem is formulated as a joint optimization, which is then solved by a dynamic algorithm and is potentially able to achieve the optimum solution under certain assumptions. A specific simplified scenario is also evaluated, where the scheduling is actually an inherently NP-Hard problem, and thus a heuristic is proposed and compared with optimum results, when feasible. The channel loading and scheduling approach is somewhat similar to the one proposed in [28] but this chapter focuses on a throughput optimal approach [14], is inherently multi-hop, and congestion control is also integrated in the framework. We build on the past work on network utility optimization, by using the notion of *virtual links* to facilitate analysis of multi-channel networks.

3.1 Introduction

New challenges in wireless network design refer to a more efficient bandwidth utilization and the use of new networking paradigms. The former goal is related to the growing bandwidth demand and the scarcity of available spectrum. The latter refers to the need for flexible and easy deployment, self configuration and adaptation to the working condition. Multi-hop wireless networks have been identified as a valuable networking paradigm able to fulfill the previous requirements. Examples of multi-hop wireless networks include ad hoc networks and mesh networks. Practical interest in multi-hop wireless networks is confirmed by the recent development of standards which explicitly encompass the mesh paradigm, where the backhaul network is organized in an ad hoc topology. The IEEE 802.16 standard [3] is one example. In the context of 802.11 networks a special working group is dedicated to the mesh extension, which is referred to as 802.11s [4]. Other standardization efforts are focusing on the introduction of mesh-like support in their network architecture, such as 802.15.3/4, where the network architecture implicitly supports a mesh-like structure, and 802.15.5, which is working to define a mesh structure for personal area networks. It is clear that a deep understanding and the ability to optimize the performance of multi-hop wireless networks will offer significant benefits in these contexts.

In the last few years great attention has been devoted to networks where each node is provided with multiple radio interfaces and multiple channels are available [99]. In this scenario, each radio interface can be tuned on a different channel and cultiple concurrent transmission are possible. This approach is particularly interesting if applied to 802.11 networks, since multiple channels are already available and devices provided with multiple wireless networking cards are being designed and already exist in some testbeds.

A lot of effort has also been spent in the last few years to understand the challenges related to resource allocation in such networks, where the increased number of variables to be jointly optimized represents a big issue. The problem has been approached from different perspectives, ranging from heuristic and protocol oriented solutions [100, 48, 47, 42], whose performance is far from being exactly defined, to the determination of theoretical bounds [52, 2, 28], whose practical implementation is not straightforward. It is thus worth investigating an approach aiming at the design of practical algorithms based on a solid theoretical background, which can be analytically proved to guarantee some performance bounds [28].

The aim of this paper is to provide a simple and clear framework for investigating the performance of multi-channel multi-radio networks as a function of the number of channels, interfaces and concurrent end–to–end transmissions. We consider the problem of joint congestion control, channel allocation and scheduling for multi-hop wireless networks in a general communication and interference scenario. The problem is formulated as a joint optimization, which is then solved by a dynamic algorithm. A specific simplified scenario is evaluated, where the scheduling is actually an inherently NP-Hard problem, and thus a heuristic is proposed. The channel loading and scheduling approach is somewhat similar to the one proposed in [28] but our paper focuses on a throughput optimal approach [14], is inherently multi-hop, and congestion control is also integrated in the framework. We build on past work on network utility optimization (discussed in Section 3.3), by using the notion of *virtual links* to facilitate the analysis of multi-channel networks.

The paper is organized as follows. The complete system model and the goal of the proposed analysis are presented in Section 3.2. Related work is reviewed in Section 3.3. The optimization problem is formulated in Section 3.4 and the proposed solution is presented in Section 3.5 together with stability issues, addressed in Section 3.6. The scheduler is defined in Section 3.7 and simulation results for the whole algorithm are in Section 3.8. Conclusions end the paper.

3.2 System model

Our system model is derived from similar models used in past work [28], [14] with suitable modifications to capture the availability of multiple channels, as described below.

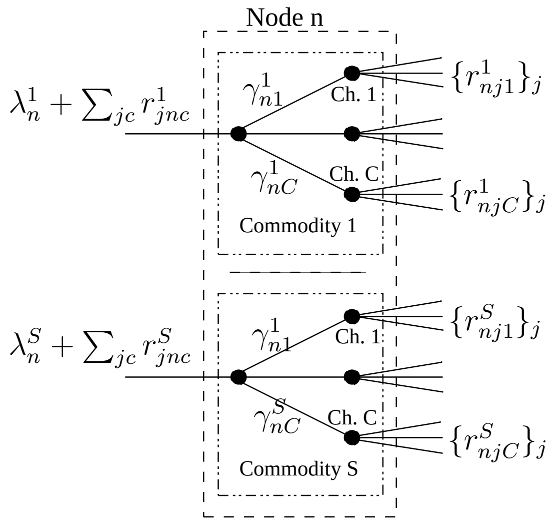

Figure 3.1: Node model

Consider a multi-hop wireless network. Each node in $\{n : n = 1, \ldots, N\}$ is provided with I_n half duplex wireless interfaces. At any given time, each interface can tune to any one of C channels $\{c : c = 1, \ldots, C\}$. The channel used by an interface may change over time. For the algorithm definition, a general interference model is initially assumed (which can also encompass non-orthogonal channels). In Section 3.7, a simplified interference model based on orthogonal channels, and communication and interference graphs, is used in order to define a greedy heuristic.

Traffic flows are, in general, carried over multi-hop routes. Each set of flows with the same destination will be referred to as a single *commodity* in the following. Let $\{s : s = 1, \ldots, S\}$ be the commodities set. The input rate for commodity s at node n is λ_n^s. Let $\overline{\lambda}$ be the vector of all input rates. Each input rate can assume values $\lambda_n^s \in \Lambda_n^s$.

As a result of the proposed algorithm, each node n will be provided with an input queue $U_{n,in}^s$ for each commodity s, and $C \times S$ output queues, $U_{n,c,out}^s$ one for each channel-commodity pair. All the incoming traffic for commodity s is loaded on queue $U_{n,in}^s$. Output queues for commodity s are loaded using packets stored in queue $U_{n,in}^s$, according to the policy described in the next sections. Inside each node, and for each commodity, a connection is defined between the input queue $U_{n,in}^s$ and each of the output queues $U_{n,c,out}^s$ on different channels, for the same commodity. Such connections will be referred to as *virtual links* in the following. Let $\gamma^s_{n,c}$ be the rate at which data is transferred from the input queue $U^s_{n,in}$ to the output queue $U^s_{n,c,out}$, i.e., the rate of the associated virtual link. Let $\overline{\gamma}$ denote the vector for all $\gamma^s_{n,c}$ and Ψ its feasible set, which represents the rate region for the virtual links. The set Ψ will be defined in Section 3.6, based on a stability argument and in order not to modify the capacity region of the actual network.

Let $r^s_{a,b,c}$ be the transmission rate associated with the flow between nodes a and b on channel c, carrying traffic for commodity s, and let $\overline{r}$ be the corresponding vector for all a,b and c. The physical layer capacity for the link between nodes a and b on channel c is denoted as $w_{a,b,c}$. Let us denote by $\overline{w}$ the vector consisting of $w_{a,b,c}$ for all nodes a, b and channel c. The feasible rate region, i.e., the set of all feasible $\overline{w}$ vectors, is denoted as $\mathcal{W}$, which depends on the interference model, and, in general, is also constrained by the limited number of wireless interfaces at each node.

The utility function for commodity s associated with each source node n is denoted by $G^s_n(\lambda^s_n)$. To allow the use of convex optimization techniques, all the utility functions are assumed to be strictly concave, and the rate vectors $\overline{w}$ will actually be considered as belonging to the convex hull of the set $\mathcal{W}$, $\overline{w} \in$ Co($\mathcal{W}$). Similar assumptions have been made in past work as well [20, 14].

The goal of the proposed algorithm is to jointly define

- congestion control
- routing
- channel loading
- interface binding and scheduling

with provable properties in terms of stability (achieved when the following property is satisfied: $\lim_{t\to\infty} E[\sum_{n,c,s}(U^s_{n,c,out} + U^s_{n,in})] < +\infty$) and network utility maximization.

The use of multiple queues, similarly to [28], has been chosen in order to exploit multichannel weighted matching algorithms in the scheduling operation. The algorithm presented in [1], which is based on non-weighted matchings, can also be adapted to fit in our framework. As will be clear in the following, in case $C = I$ (e.g., OFDMA systems) the scheduling problem turns out to be decoupled along channels. Suboptimal heuristics could also take advantage of individual channel queue length information.

In the following, a general formulation is presented in terms of an optimization problem on a network flow. A Lagrangian relaxation allows to define a distributed utility maximization, channel loading and scheduling based on the concept of “backpressure” [14]. Our approach makes use of “virtual links” for loading the queues on each channel. A stability issue in the definition of virtual link rates is discussed below, and a Lyapunov argument is used to justify the solution. A heuristic way to solve the scheduling optimization is also discussed for the case of a simplified transmission and interference model. A lower bound for the performance of the joint algorithm is also identified later in the paper.

3.3 Related work

The concept of "layering as optimization decomposition" has been investigated in the last few years as a powerful way to analytically define cross-layer optimization problems and at the same time design feasible algorithms for their solution [20]. In particular, joint algorithms for congestion control and transmission scheduling have been proposed [12, 101] which are able to jointly optimize source rate and link scheduling [14], [13], [15] including also the power control operation [16, 17]. The mathematical tools widely used in this new approach are essentially optimization problem decomposition by Lagrange relaxation, sub gradient algorithms and Lyapunov stability [18, 19]. The work presented in this paper is based on the decomposition of an optimization problem defined over the multi-channel network model.

The solution is related to the general scheduling algorithm presented in [14, 16]. Given a set of input rates which lies inside the capacity region of the system, this algorithm is able to guarantee stability (i.e., bounded queue lengths). The core of the scheduler is based on the maximization of a metric which depends on the rate allocated to each link, multiplied by the difference between the queue length at the link receiver side minus the queue length at the transmitter side (thus the name "backpressure"). In [14], a congestion controller is added on top of the scheduling algorithm which is proved to converge to a solution close to the optimum.

The use of an imperfect scheduler in the joint scheduling and congestion control may in general lead to poor performance [23]. In case an imperfect scheduler is used, the joint algorithm presented in [14] is proved to be able to guarantee stability within a capacity region scaled by a factor which depends on the imperfect scheduler. This opens the way to the implementation of reduced complexity schedulers. In case a "protocol interference" model is considered, the scheduling, for a single channel scenario, becomes a weighted maximum independent set problem. The problem is in general NP-hard [27]. Clearly, a greedy centralized algorithm which selects at each step the link with the highest metric and discards all the interfering links can achieve a capacity region reduced by a factor of $1/K$ where K is the interference degree [23]. In [29], it is pointed out that such a greedy approach is optimal in graphs with particular structure.

Algorithms based on a maximal independent set scheduler (non weighted) are known for single hop networks and are presented in [33, 34], but this approach can not be extended to the multihop case. In this case a different scheduler has to be used, which exploits additional information on the traffic intensity or number of hops [35].

A novel approach for the scheduling problem is also proposed in [1], where a randomized algorithm is used. The problem of maximizing the backpressure function is converted to the problem of comparing the backpressure value obtained in subsequent random schedules. At each time slot, the backpressure achieved by a new random maximal matching is compared with the one achieved by the previous schedule. The best schedule is applied. A distributed algorithm is also presented.

The closest work for multi-channel multi-radio wireless networks is the one in [28]. The authors propose a channel loading mechanism which, combined with a multi-channel maximal scheduler, is able to keep the network stable inside a subset of the capacity region. The network model is such that each node is provided with an input queue for each commodity and an output queue for each commodity-channel pair. A known traffic rate is applied at each input queue for each node and, based on a metric accounting for the queue lengths of all interfering nodes, a channel loading policy is defined. A multi-channel maximal scheduler is then applied to schedule the backlogged links. This approach is extended to the multihop case only for the case where information on the source rate is available and a congestion control is not considered.

An optimization approach is also used in [2], where an LP network flow problem is defined to model routing and channel loading. The solution is used to obtain an upper bound for the performance. A greedy scheduler based on the outcome of the previous LP solution is then applied for solving the actual resource allocation. A similar analysis is also found in [52].

In this paper, a network structure similar to the one in [28] is assumed, and the optimization approach is based on the decomposition presented in [20] and the argumentation in [14]. We propose an algorithm that works in a multihop scenario, and whose simple channel loading mechanism is based only on local information. The complexity is moved to the scheduling operation, which in general can be very complex. The proposed algorithm makes use of one input queue at each node for each commodity and one output queue for each channel–commodity pair at each node. The queue lengths are used, at each time step, to make dynamic decisions about congestion control, channel loading and transmission scheduling. In particular, "virtual links" are introduced in order to model the channel loading operation. The algorithm is analytically formulated and then tested by simulation in a simplified communication and interference scenario. The impact of the number of channels, interfaces and commodities on the network performance is investigated.

3.4 Formulation as an optimization problem

The goal of the proposed algorithm is to solve the following optimization problem (see Table 3.1 for a summary of the symbol definitions):

$$\max_{\overline{\lambda},\overline{r},\overline{w},\overline{\gamma}} \sum_{n,s} G_n^s(\lambda_n^s) \tag{3.1}$$

s.t.:

$$\sum_{i,c} r_{i,n,c}^s + \lambda_n^s \leq \sum_c \gamma_{n,c}^s \ \forall n, s \tag{3.2}$$

$$\gamma_{n,c}^s \leq \sum_j r_{n,j,c}^s \ \forall n, c, s \tag{3.3}$$

$$\sum_s r_{i,n,c}^s \leq w_{i,n,c} \ \forall i, n, c \tag{3.4}$$

$$\overline{\gamma} \in Co(\Psi) \tag{3.5}$$

$$\overline{w} \in Co(\mathcal{W}) \tag{3.6}$$

$$\lambda_n^s \in \Lambda_n^s \ \forall n, s \tag{3.7}$$

In the previous model:

- (3.1) is the objective function
- (3.2) is the flow conservation constraint at the input of each node
- (3.3) is the flow conservation constraint at the output of each node
- (3.4) is the constraint that the aggregate flow on a link must not exceed the physical rate
- (3.5) is the constraint on the flow in the virtual links for the channel loading: this will be specified to model different requirements. Note the convex hull operator.
- (3.6) is the feasible rate region for the actual links.
- (3.7) is the feasible set for the input rates.

Based on the assumption on the utility functions and on the convexity of the domain, (3.1)-(3.7) is a convex optimization problem.

3.5 Network flow optimization

The solution of the optimization problem is obtained via its dual problem, relaxing all the constraints (3.2) and (3.3). The procedure is based on prior work in [19] and [14].

Symbols:	
$G_n^s(\lambda_n^s)$	Utility function
$\overline{\lambda} = [\lambda_n^s]$	Injected input rate
$\overline{r} = [r_{a,b,c}^s]$	Flows associated to channel-link-commodity connections
$\overline{\gamma} = [\gamma_{n,c}^s]$	Flows that load output channel-commodity queues
$\overline{w} = [w_{a,b,c}]$	Physical rates associated to physical channel-link
Ψ	Feasible "virtual rate region" for channel loading
$\mathcal{W}$	Feasible rate region for actual physical links
Λ_n^s	Feasible input rates

Table 3.1: Symbols

Let $\mathbf{U_{in}} = [U_{n,in}^s]$ and $\mathbf{U_{out}} = [U_{n,c,out}^s]$ be the vectors for all the Lagrange multipliers associated to constraints (3.2) and (3.3) respectively. Let $\mathbf{U} = [\mathbf{U_{in}}, \mathbf{U_{out}}]$ be the vector for all the Lagrange multipliers.

Relaxing the constraints (3.2) and (3.3), the Lagrange dual function for the problem is:

$$\begin{aligned} L(\mathbf{U}) &= \max_{\overline{\lambda},\overline{r},\overline{w},\overline{\gamma}} \left\{ \sum_{n,s} G_n^s(\lambda_n^s) + \right. \\ &+ \sum_{n,s} U_{n,in}^s \left(-\sum_{j,c} r_{j,n,c}^s - \lambda_n^s + \sum_c \gamma_{n,c}^s \right) + \\ &+ \left. \sum_{s,c,n} U_{n,c,out}^s \left(-\gamma_{n,c}^s + \sum_j r_{n,j,c}^s \right) \right\}, \end{aligned}$$

where the optimization variables $\overline{\lambda}, \overline{r}, \overline{w}, \overline{\gamma}$ are still subject to constraints (3.4)-(3.7) (here, and in the following, constraints are omitted to simplify notation).

The previous expression can be rewritten as:

$$L(\mathbf{U}) = \max_{\overline{\lambda}} \left\{ \sum_{n,s} G_n^s(\lambda_n^s) - \lambda_n^s U_{n,in}^s \right\} + \tag{3.8}$$

$$+ \max_{\overline{r},\overline{w}} \left\{ \sum_{i,j,s,c} \left(U_{i,c,out}^s - U_{j,in}^s \right) r_{i,j,c}^s \right\} + \tag{3.9}$$

$$+ \max_{\overline{\gamma}} \left\{ \sum_{s,n,c} \left(U_{n,in}^s - U_{n,c,out}^s \right) \gamma_{n,c}^s \right\}. \tag{3.10}$$

Note how each maximization represents a different "layer" in the optimization task:

- (3.8) congestion control;

- (3.9) flow allocation, routing and physical rate allocation;
- (3.10) channel management (stability, channel loading, ...)

Let $\tilde{\lambda}(\mathbf{U})$, $\tilde{r}(\mathbf{U})$, $\tilde{w}(\mathbf{U})$, $\tilde{\gamma}(\mathbf{U})$ be the vectors of optimum values for a given set of Lagrange multipliers, that clearly depend on $\mathbf{U}$. The optimizations in (3.8) and (3.10) can be carried out based only on local information, and are thus suitable for a distributed implementation. The optimization of $\tilde{r}$ and $\tilde{w}$ in (3.9) instead requires the knowledge of the feasible rate region $\mathcal{W}$ and could in general be solved by using a centralized algorithm, even though distributed solutions can be developed under particular interference assumptions, such as for example in [1]. In particular, in order to optimize (3.9), for each link between nodes i and j on channel c define $s^* = \arg\max_s \left\{ \left(U^s_{j,c,out} - U^s_{i,in} \right) \right\}$. The flow allocation is given by setting $r^{s^*}_{i,j,c} = w_{i,j,c}$ and $r^s_{i,j,c} = 0$ for $s \neq s^*$. Once the flow to be potentially loaded on a physical link has been chosen, the following maximization has to be performed: $\tilde{w} = \arg\max_{\overline{w}} \left\{ \sum_{i,j,c} \left[U^{s^*}_{i,c,out} - U^{s^*}_{j,in} \right]^+ w_{i,j,c} \right\}$. This is the backpressure algorithm [14].

In Section 3.7, an assumption about a specific feasible rate region will be discussed, and the design of a greedy algorithm to compute $\tilde{w}$ will be presented, together with a lower bound performance index.

Suppose for the moment that the only constraint imposed to the virtual link rates is:

$$\sum_c \gamma^s_{nc} < \Gamma^s_n$$

were each Γ^s_n is a constant, which is set according to the stability and capacity preservation criterion discussed in Section 3.6.

Under this assumption, the maximization in (3.10) requires that for each node n and commodity s, $c^* = \arg\max_c \left\{ \left(U^s_{n,in} - U^s_{n,c,out} \right) \right\}$ is chosen. If $\left(U^s_{n,in} - U^s_{n,c^*,out} \right) > 0$ then set $\gamma^s_{n,c^*} = \Gamma^s_n$ and all $\gamma^s_{n,c} = 0$ for $c \neq c^*$, else set all $\gamma^s_{n,c} = 0 \; \forall c$ (so that the summation is 0; otherwise the summation would be negative). This is essentially the backpressure based algorithm for the virtual links $\overline{\gamma}$.

The Lagrange function is convex, thus the multipliers can be computed using a sub gradient algorithm. It is known that a sub gradient for a given vector of Lagrange multipliers is the vector consisting of all multiplicative terms in the Lagrange function. Note that such multiplicative terms are the results of maximizations (3.8)–(3.10). With this choice, the Lagrange multipliers are computed using a sequential algorithm which, at each step, updates them based on the value of the local sub gradient. Let t be the iteration index, which can be associated with a time-slot in the system evolution. Thus the updating rules for each multiplier at time $t+1$ are:

$$\begin{aligned} U^s_{n,in}(t+1) = \Big[& U^s_{n,in}(t) + \alpha_1(\tilde{\lambda}^s_n(\mathbf{U}(t)) + \\ & + \sum_{j,c} \tilde{r}^s_{j,n,c}(\mathbf{U}(t)) - \sum_c \tilde{\gamma}^s_{n,c}(\mathbf{U}(t))) \Big]^+ \end{aligned} \tag{3.11}$$

$$U^s_{n,c,out}(t+1) = \Big[U^s_{n,c,out}(t) + \alpha_2(\tilde{\gamma}^s_{n,c}(\mathbf{U}(t)) + \\ - \sum_j \tilde{r}^s_{n,j,c}(\mathbf{U}(t)))\Big]^+ . \quad (3.12)$$

In order to get a solution which converges to a stable value, α_1 and α_2 should be set to be small constants, whereas in the case $\alpha_1 = \alpha_2 = 1$ the solution will exhibit an oscillatory behavior around the convergence point.

However if $\alpha_1 = \alpha_2 = 1$ the Lagrange multipliers obey the same dynamic equation as the queue lengths. This makes this case most interesting from a practical perspective, since the optimization can be performed by just measuring the queue lengths and using these values directly in the algorithm. For this reason, we will focus on this case in the following.

Note that at each time t a new sub gradient has to be computed, thus the optimizations (3.8)–(3.10) has to be repeated at each time slot. Let $\tilde{\lambda}(t)$, $\tilde{r}(t)$, $\tilde{w}(t)$, $\tilde{\gamma}(t)$ denote the vectors solutions of the optimization variables where the time index has been explicitly shown, whereas $\mathbf{U}$ is neglected to simplify notation.

Based on the previous argumentation, the proposed algorithm for joint congestion control, channel allocation and scheduling is presented in Algorithm 1.

3.6 Channel Loading: Stability by Lyapunov drift

In the previous section the feasible rate set for the virtual links used for the channel loading has not been specified. Here it is proved that a sufficient condition for stability requires the aggregated rate of the virtual links, used for the channel loading, to be bounded.

The stability of the system is derived using a Lyapunov argumentation. Consider the input rate for all commodities as fixed (no congestion control) and assume it falls within the capacity region of the network.

The considered Lyapunov function is $L = \sum_{n,s}(U^s_{n,in})^2 + \sum_{n,s,c}(U^s_{n,c,out})^2$ and the proof is derived from the one in [14] Sec. 4.2.

Consider the queue updating rules (3.11), (3.12) with $\alpha_1 = \alpha_2 = 1$. The drift associated to the Lyapunov function L is denoted with $\Delta(\mathbf{U}(t)) = E\left[L(\mathbf{U}(t+1)) - L(\mathbf{U}(t)) | \mathbf{U}(t)\right]$

Algorithm 1 Joint optimization

At each time step t, perform the following operations.

1. Congestion control. For each commodity s and node n:
   $\tilde{\lambda}_n^s(t) = \sup_{\{\lambda_n^s \in \Lambda_n^s\}} \left\{ G_n^s(\lambda_n^s) - \lambda_n^s U_{n,in}^s(t) \right\}$
2. Channel allocation. For each commodity s and node n:
   $c^* = \arg\max_c \left\{ \left(U_{n,in}^s(t) - U_{n,c,out}^s(t) \right) \right\}$,
   if $\left(U_{n,in}^s(t) - U_{n,c^*,out}^s(t) \right) > 0$ **then**
   set $\tilde{\gamma}_{n,c^*}^s(t) = \Gamma_n^s$ and all $\tilde{\gamma}_{n,c}^s(t) = 0$ for $c \neq c^*$
   else
   set all $\tilde{\gamma}_{n,c}^s(t) = 0 \; \forall c$.
   end if
3. Scheduling and routing. For each link between nodes i and j on channel c:
   $s^* = \arg\max_s \left\{ \left(U_{i,c,out}^s(t) - U_{j,in}^s(t) \right) \right\}$.
   $\tilde{w} = \arg\max_{\overline{w}} \left\{ \sum_{i,j,c} \left[U_{i,c,out}^{s^*}(t) - U_{j,in}^{s^*}(t) \right]^+ w_{i,j,c} \right\}$
   if $\left(U_{i,c,out}^{s^*}(t) - U_{j,in}^{s^*}(t) \right) > 0$ **then**
   set $\tilde{r}_{i,j,c}^{s^*}(t) = \tilde{w}_{i,j,c}(t)$ and all $\tilde{r}_{i,j,c}^s(t) = 0$ for $s \neq s^*$
   else
   set all $\tilde{r}_{i,j,c}^s(t) = 0 \; \forall s$
   end if
4. Queues update:
   the queues are automatically updated according to the rules in (3.11) and (3.12) with $\alpha_1 = \alpha_2 = 1$.

and can be easily bounded as

$$\Delta(\mathbf{U}(t)) \leq B + 2 \sum_{ns} U_{n,in}^s(t) E\left[\lambda_n^s(t)\right] + \tag{3.13}$$

$$-2E\left[\sum_{i,j,c,s} r_{i,j,c}^s [U_{i,c,out}^s(t) - U_{j,in}^s(t)] \mid \mathbf{U}(t) \right] + \tag{3.14}$$

$$-2E\left[\sum_{c,n,s} \gamma_{n,c}^s [U_{n,in}^s(t) - U_{n,c,out}^s(t)] \mid \mathbf{U}(t) \right] + \tag{3.15}$$

$$+ \sum_{n,s} \left(\sum_c \gamma_{n,c}^s \right)^2 + \sum_{n,c,s} (\gamma_{n,c}^s)^2 , \tag{3.16}$$

where B is a constant term depending on the r terms.

According to Corollary 3.9 in [14], if the input rate λ_n^s (which is loaded only on the input queue) is such that $\lambda_n^s + \epsilon\ \forall n, s$ (for a small ϵ) lies inside the capacity region, then there exists a randomized scheduling $\hat{r}$, $\hat{w}$ and $\hat{\gamma}$, such that

$$E\left[\sum_c \hat{\gamma}_{n,c}^s - \sum_{i,c} \hat{r}_{i,n,c}^s\right] = \epsilon + \lambda_n^s\ \forall n, s, c$$
$$E\left[\sum_j \hat{r}_{n,j,c}^s - \hat{\gamma}_{n,c}^s\right] = 0\ \forall n, s$$

and thus choosing a schedule $\tilde{r}, \tilde{w}, \tilde{\gamma}$ according to the maximization in (3.9) and (3.10), then (3.14) and (3.15) can be bounded leading to

$$\begin{aligned}\Delta(\mathbf{U}(t)) \leq B' - 2\epsilon \sum_{n,s} U_{n,in}^s + \\ + \sum_{n,s}\left(\sum_c \gamma_{n,c}^s\right)^2 + \sum_{n,c,s}(\gamma_{n,c}^s)^2.\end{aligned} \tag{3.17}$$

Note that if $\sum_{n,s}(\sum_c \gamma_{n,c}^s)^2 + \sum_{n,c,s}(\gamma_{n,c}^s)^2$ is bounded, the drift becomes negative as the queue lengths increase above a given threshold. Thus for instance we can define the feasible rate for the virtual link as $\Psi = \{\gamma_{n,c}^s : \sum_c \gamma_{n,c}^s < \Gamma_n^s\ \forall s, n\}$, with Γ_n^s suitable positive constants. The proposed network model artificially adds the virtual links to the original network structure, thus we have to make sure the resulting network is able to provide the same capacity region as the original one. To guarantee such a property, a value for each Γ_n^s can be chosen as the smallest value greater than the maximum possible output rate for a node (which is bounded).

3.7 Scheduling

The scheduler defined in general terms in the previous section requires a centralized optimization.

Here a specific communication and interference model is considered; each node can communicate with any other node within a distance R_c; for a correct reception no concurrent transmissions are allowed within a distance R_i from the receiver and the transmitter. In this case, the problem of scheduling non interfering links while maximizing (3.8) is equivalent to finding the maximum weighted independent set in a weighted graph. Additional constraints imposed by the reduced number of interfaces have to be considered. Note that the maximum weighted independent set problem (defined for the single channel case) is known to be NP-Hard [27].

It is known [28, 23] that a greedy sequential and centralized algorithm that at each step selects the link with the highest metric and drops all the interfering links can reach at least a fraction $\beta = 1/K$ of the maximum value in (3.9), where K is the maximum number of links that cannot be scheduled because a given link has already been scheduled. In fact, at each step at most K additional links with a weight equal to or smaller than the selected one could have been scheduled. Note that the constraint on the number of interfaces can cause a link activation to prevent the use of other links in different channels. Thus K depends on the topology and on the number of channels and interfaces. This greedy scheduler allows for the solution of the whole optimization problem to converge to the optimum referred to a capacity region scaled by a factor β [14].

The previous lower bound is very conservative and the actual performance of the greedy procedure is expected to be much better than stated above. Some reasons are listed below: i) the number of contending nodes is actually only the number of backlogged nodes with positive backpressure, thus, as long as the network is not heavily loaded, this number is much smaller than K; ii) the loss in optimality β is itself a lower bound, as it assumes that each time a link is scheduled, all the dropped links have a weight which is close to the one of the scheduled link; iii) the maximal scheduling is close to the maximum scheduling in most practical topologies [29].

In the following, the proposed cross-layer algorithm has been tested using such a greedy centralized scheduler for a given topology.

3.8 Simulation results

In this section the whole algorithm is tested using the greedy centralized scheduler previously described. The presented framework allows for an extensive evaluation of the performance of a multi-channel multi-radio network as a function of several parameters, i.e., number of channels C, number of interfaces per node I, and number of commodities S. In particular, the number of interfaces is the most interesting parameter, since it represents the peculiar feature of this kind of networks, and has been the focus of previous theoretical results. In all the scenarios, the capacity of each interference free scheduled link is fixed to $1/C$ for a fair comparison among scenarios with different numbers of channels. The utility function is the same for all the nodes and is defined as $G_n^s(x) = log(x)$, which implements a fairness based congestion control. Simulations have been performed by using Matlab.

3.8.1 Comparison with optimum solution and algorithm from [1]

As previously stated, to find the optimum solution of the scheduling problem is a very complex task. In this section, the heuristic proposed in Section 3.7 is compared with the optimum solution and with the schedule provided by an adaptation to the current framework of the randomized algorithm proposed in [1].

The optimum solution is achieved by an exhaustive search on the solution tree realized through a depth-first search algorithm.

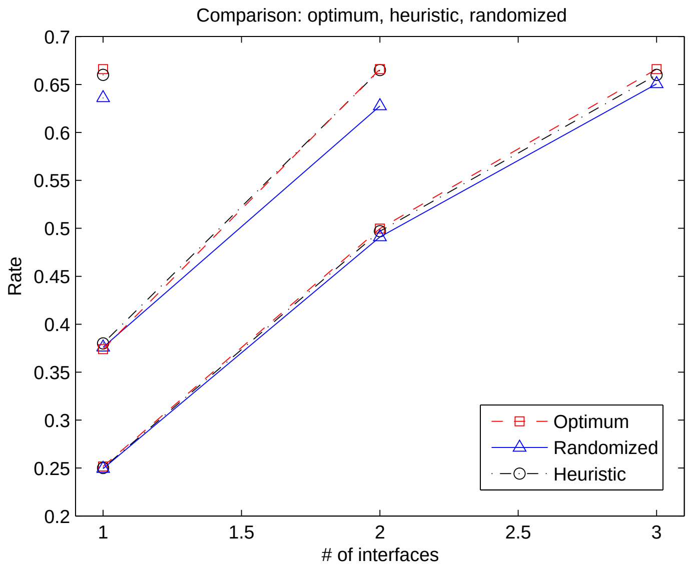

Figure 3.2: Comparison between solutions with optimum and heuristic scheduler. Dashed: optimum; dot/dashed: heuristic from Section 3.7; solid: randomized algorithm from [1]. The three sets of curves are referred to $C = 1, 2, 3$.

Most of the complexity is due to the need for searching the optimum weighed maximum matching at each time slot. Recently, in [1] a new approach to solve the scheduling problem has been proposed. At each time slot, the algorithm computes i) the backpressure achieved by using the link allocation of previous time slot and ii) the backpressure achieved by using a new tentative random allocation. The tentative allocation is a random choice of feasible communication links, which is only required to satisfy the interference and number of channel constraints. The two backpressure values are compared. If the one associated with the new tentative allocation is greater than or equal to the one achieved by the old allocation, then the tentative allocation is applied and used for the current time slot. Otherwise, the old allocation is used in the current time slot. This algorithm allows to use a maximal matching as the scheduling algorithm, instead of the maximum weighted matching required by the formulation related to the classic backpressure maximization. The algorithm requires a comparison among two values of backpressure, which have to be obtained by gathering information among nodes through message exchange; a distributed mechanism can be designed for performing this operation. It has been proved that this mechanism allows the cross-layer algorithm to reach a solution arbitrarily close to the optimum one. The logarithmic utility function has been considered in this case. The parameter determining the solution optimality is the M in (3.17). The greater M the better the solution, at the cost of an increased convergence time.

The first considered scenario is a regular linear deployment of 6 nodes, where each node communicates and interferes with the nearest neighbors. Two commodities are considered (node 1 sends traffic to node 6 and node 2 sends traffic to node 5, with IDs following the topological order). Even in this simple scenario only a very small number of channels could be tested by using the optimum algorithm.

In Figure 3.2 the aggregated received throughput is shown for each of the algorithms. In this simple scenario, the heuristic solution presented in Section 3.7 is able to reach the optimum one. The randomized algorithm and the heuristic one have been run with $M = 10$; for this value of M the heuristic solution is slightly closer to the optimum. Clearly, increasing M the randomized algorithm is proved to reach the optimum solution.

In the second scenario, a random deployment of nodes is considered, such that each node has 3 neighbors (interferers) on the average. The cases $S \in \{1, 4\}$, $C \in \{1, 2, 4, 6\}$ have been considered and results are averaged over 10 topologies. In this case the optimum solution is not achievable in reasonable time.

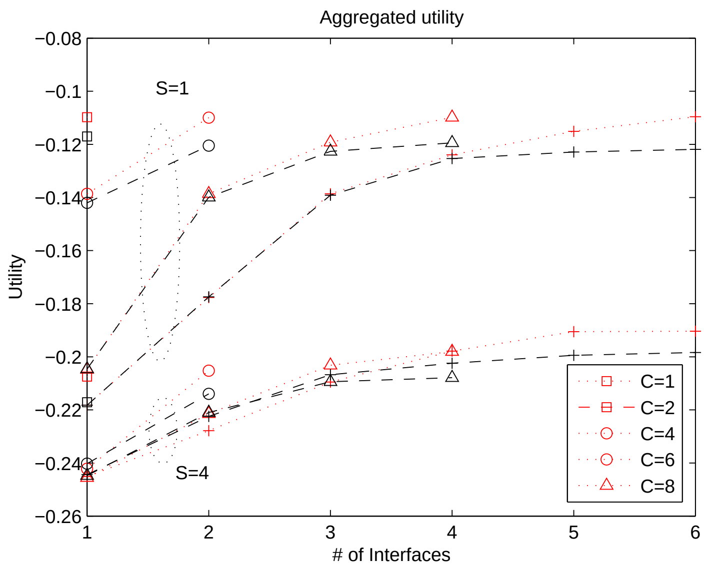

Figure 3.3: Comparison between solutions in Section 3.7 and the randomized algorithm derived from [1] . $M = 10$. Dotted: Section 3.7 ; dashed: [1].

As can be seen in Figure 3.3, the solution obtained by using the heuristic in Section 3.7 is better than the one obtained by using the algorithm in [1]. It has been verified, confirming the theory, that increasing the parameter M the utility achieved by using the algorithm in [1] increases its value for high numbers of interfaces, approaching the heuristic one, at the cost of a slower convergence.

Since the algorithm in [1] is proved to converge to the optimum solution by increasing M, this confirms that the chosen heuristic is a good approximation of the optimum solution and has better convergence properties in the considered scenario. Therefore, all other results will be provided only for the heuristic solution.

3.8.2 Grid topology

The algorithm has been simulated in a single network snapshot composed by $N = 16$ nodes, placed in a regular mesh with a distance of 0.2 units between adjacent nodes.

In this scenario $R_i = R_c = 0.3$. The system has been tested with $C \in \{1, 2, 4, 8\}$, $I \in \{1, ..., C\}$, $S \in \{1, 2, 4\}$.

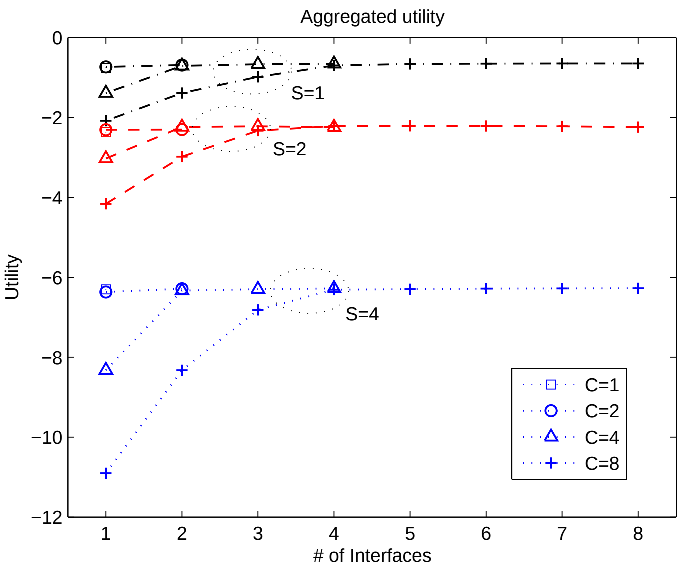

Figure 3.4: Total utility for different numbers of channels, interfaces and commodities.

As can be seen from Figure 3.4, in all cases the aggregated utility increases as the number of interfaces increases. Anyway, the additional utility gained adding a new card decreases as the number of cards increases. For instance, in case of $C = 8$, only 4 interfaces are enough for achieving the maximum utility. This is in accordance with the asymptotic analysis presented in [36]. The utility is negative because a logarithmic function is used and, in the simulated scenario, a normalized bandwidth value has been considered. Moreover the utility decreases as the number of concurrent flows increases. This is due to the specific scenario where the rate experienced by a single flow decreases as the number of flows increases due to the sharing of the medium.

Even though throughput maximization is not the main goal of the simulated algorithm, in Figure 3.5 the aggregated transmission rate of all commodities is shown. Similarly to the utility behavior, the aggregated rate increases as the number of interfaces increases and the maximum value is reached using a number of interfaces smaller than the number of channels. As the number of commodities increases, the aggregated rate increases, showing that the spatial reuse of the medium is exploited.

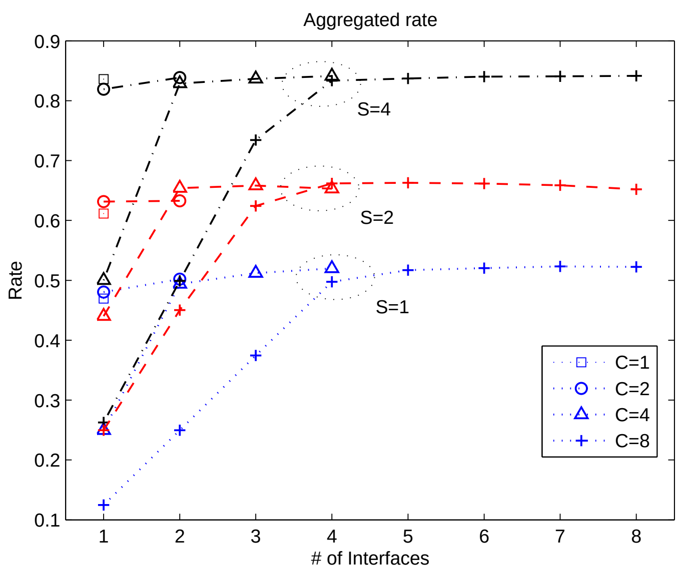

Figure 3.5: Aggregated transmission rate for different numbers of channels, interfaces and commodities.

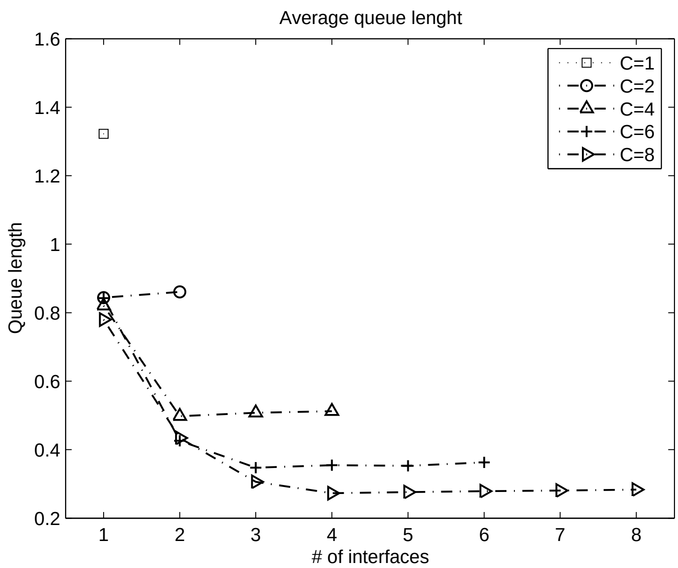

Figure 3.6: Average queue length for different numbers of channels and interfaces. $S = 4$.

In Figure 3.6 the average queue length in the stationary regime is shown as a function of the number of interfaces and channels. As the number of interfaces increases the average length decreases. This has an impact on the end-to-end delay, which becomes smaller if a higher number of interfaces is used. The queue length decreases also as the number of channels increases. Note in particular that in all cases $I = 4$, $C \in \{4, 8\}$ the maximum throughput is reached (see Figure 3.5). On the other hand a higher number of channels allows for a reduced queue length and thus a reduced delay.

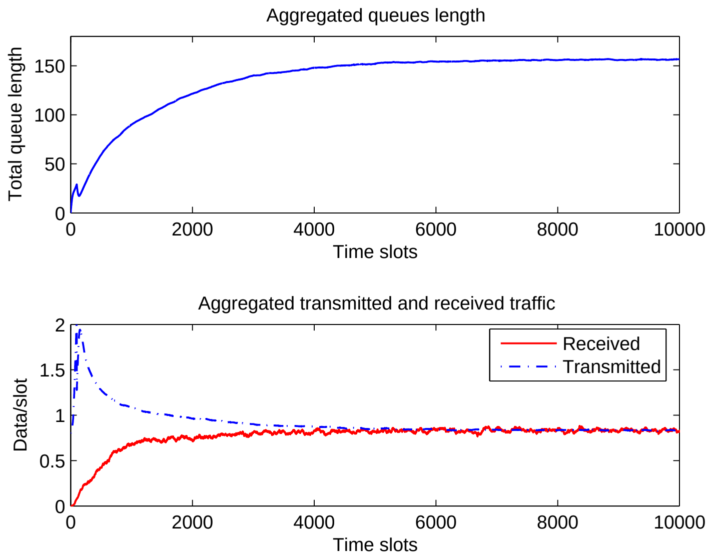

Figure 3.7: System time evolution. The curves shown are averaged over a moving window of 100 samples. $C = 8$, $S = 4$, $I = 4$.

The proposed algorithm has been proved to asymptotically converge to the solution of the joint resource allocation problem, but the proposed analysis gives no insight on the time required for the algorithm to converge. Figure 3.7 shows a typical trend for the time evolution of aggregated queue lengths, aggregated transmitted rate by the sources and aggregated received rate at the sinks. As can be seen, the convergence is reached after a relatively high number of iterations.

A more exhaustive investigation of the convergence time is presented in Figure 3.8 where the time needed to reach a stationary condition is plotted for different number of interfaces, channels, and commodities. As can be seen, the convergence time decreases as the number of interfaces increases, and increases as the number of channels or commodities increases.

Our interpretation for this behavior is that, as the number of queues in the system increases, more time is required for all the queues to be served and thus reach a stable configuration. This transient phase could be interpreted as a route discovery mechanism. Increasing the number of interfaces leads to a higher number of concurrent transmissions,

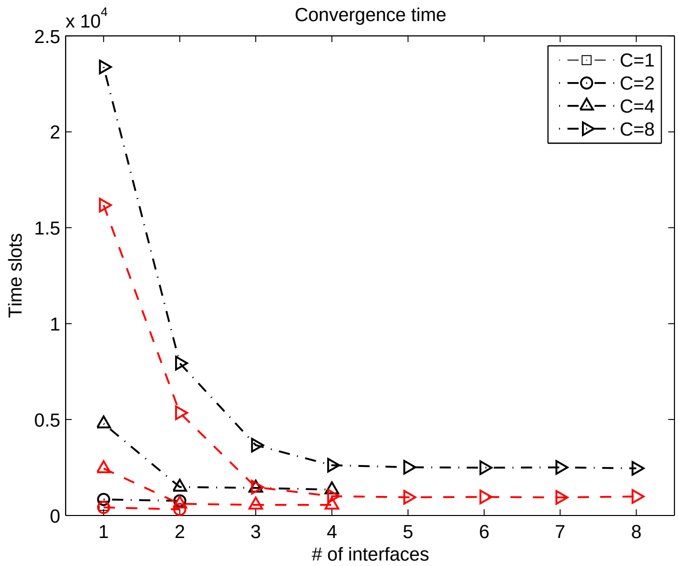

Figure 3.8: Number of slots required for convergence, measured as the number of iterations needed to reach an aggregated queue length within 10% of its steady-state value. Dashed: $S = 2$; Dash-dotted: $S = 4$.

which speeds up the convergence process.

In some cases the time required for convergence is very long. This can limit the practical implementation of such an algorithm in an actual network. A reason for the slow convergence is related to the routing mechanism, which imposes no constraints on the feasible paths for the traffic. The traffic thus can travel in all directions until a stable configuration is reached. It would be interesting to define a policy for setting a reduced number of feasible paths for each commodity. Convergence delay also depends on the particular congestion controller. A detailed investigation is out of the scope of this paper and represents an open research issue as pointed out in [20].

3.8.3 Random topology

The algorithm has also been tested using random topologies where nodes are uniformly placed in a unit square area. The presented results are averaged over 10 random topologies. Only connected topologies are considered. As in the previous case, each node can potentially communicate with all neighbors within a distance of 0.3 and, when transmitting, it causes interference to neighbors within a distance of 0.3. The average number of nodes within the communication range is defined as D.

As we are interested in the system behavior as a function of the number of interfaces, rather than in its absolute value, Figure 3.9 (Figure 3.10) shows the ratio between the experienced utility (rate) for a given set of parameters and the maximum utility (rate) achieved

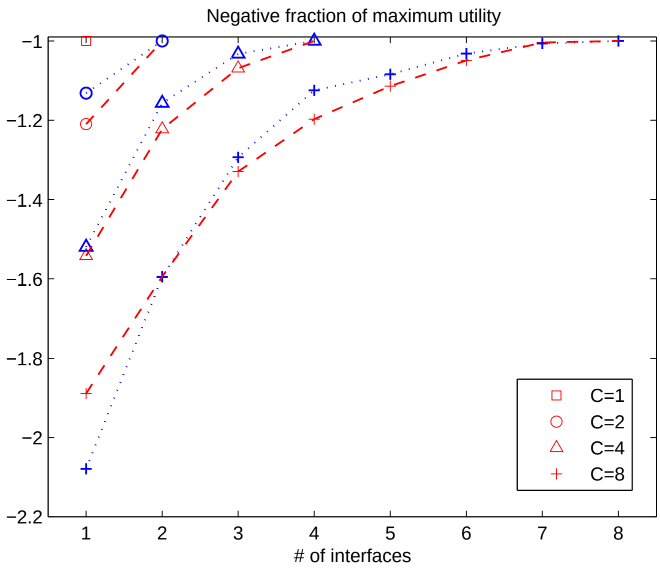

Figure 3.9: Utility normalized with respect to the maximum utility attained with the maximum number of interfaces. The negative ratio is plotted in order to provide an easier comparison with Figure 3.10. Results are averaged over three different numbers of commodities $S \in \{1, 2, 4\}$ and 10 random topologies. Dotted: $D = 7$, dashed: $D = 3$.

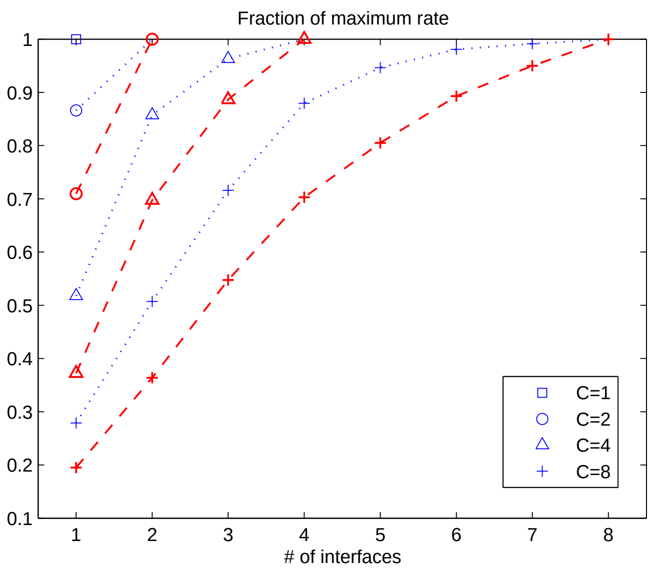

Figure 3.10: Fraction of the maximum rate. Results are averaged over three different numbers of commodities $S \in \{1, 2, 4\}$ and 10 random topologies. Dotted: $D = 7$, dashed: $D = 3$.

with the highest number of interfaces. Results are averaged over $S \in \{1, 2, 4\}$ and 10 random topologies for each value of S. Two different node densities are considered, i.e., $D = 3$ and $D = 7$.

The behavior is similar to the one already described for the grid topology. It can be noted that a higher node density allows for a reduced number of interfaces needed to reach the same utility and rate values. Once again this is in accordance with the analysis in [36].

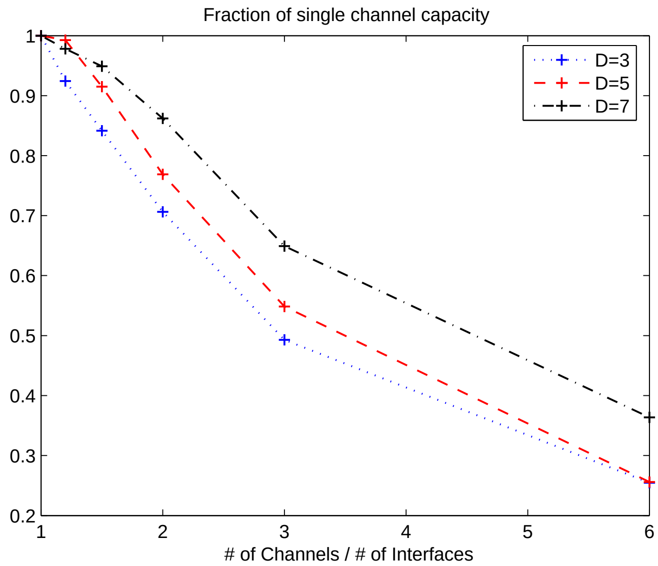

Figure 3.11: Aggregated rate, normalized with respect to the single channel case, as a function of the ratio between the number of channels and the number of interfaces. Results are referred to $C = 6$, $I \in \{1, 2, 3, 4, 5, 6\}$ and averaged over $S \in \{1, 2, 4\}$ and 10 random topologies.

In Figure 3.11 the aggregated rate, normalized with respect to the single channel case, is plotted as a function of the ratio between the number of channels and the number of interfaces. The behavior is similar to the one described in [36].

3.8.4 Comparison with the results in [2]

The scenario considered in [2] has been reproduced and a comparison between the performance of our algorithm and the one presented in [2] has been made. The algorithm in [2] formulates the resource allocation as a network flow problem and solves a linear program in order to define an upper bound on the achievable performance. Then a greedy algorithm is applied for the scheduling operations. All the sources have the same traffic requirements (no congestion control) and the objective is to find the maximum input rate for which a solution exists. Note that our algorithm aims at the utility maximization rather than at the maximization of the input rate. Nonetheless, assuming a logarithmic utility, fairness among different flows is enforced, thus pushing our input rate scenario toward the one defined in [2].

Note that the optimization in [2] uses a centralized LP solution, while our algorithm can be run in a fully distributed way, as long as a distributed scheduling mechanism is available.

A grid 5×6 topology is considered; each node has at most 4 neighbors. Four sinks for the traffic are considered ($S = 4$) and results are averaged using $\{5, 10, 15, 20, 25\}$ traffic sources. Each sink is placed on a different quadrant, and sources are connected to the closest sink. Results are shown in terms of the aggregate rate, normalized with respect to the single channel case. In this formulation each channel has a fixed capacity, so that the total bandwidth is increasing with the number of channels.

As [8] does not provide all the details of the considered scenario, in trying to reproduce it in our framework we had to make some assumptions on the positions of the source and sink nodes. Although this makes a detailed quantitative comparison difficult, it still allows to verify that the two approaches exhibit consistent behaviors. Figure 13 shows the rate gain in the two cases as a function of the number of channels and interfaces. It is clear from these results that the two approaches, though based on different techniques, have a qualitatively similar behavior. On the other hand, while the scheme in [8] is completely centralized and is more useful as a benchmark than as a practical solution, the features of our scheme can give some insight for practical implementation.

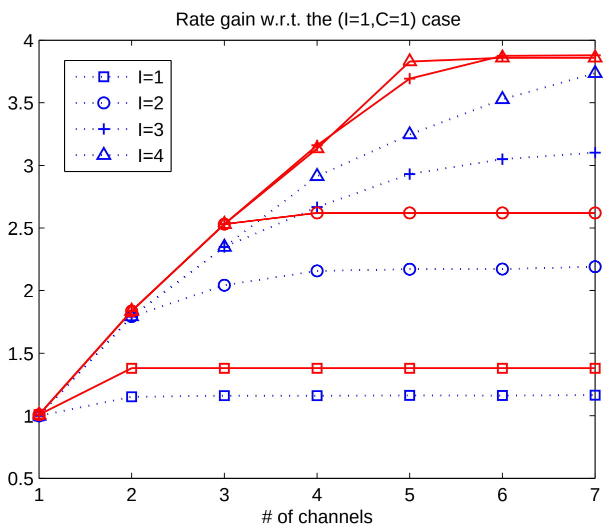

Figure 3.12: Rate gain factor with respect to the single channel case. Dotted: our algorithm. Line: copied from Figures 7 and 8 in [2]

3.9 Conclusions

A joint congestion control, channel allocation and scheduling algorithm for multi-channel multi-interface multi-hop wireless networks has been presented. The problem of maximizing a utility function of the source rate has been defined as an optimization problem and then solved by a dynamic algorithm.

The algorithm decomposes the whole optimization in different functional sub-optimizations and uses the queues length as a way to allow a joint solution of different optimization tasks.

A queue at the input of each node for each commodity and a queue at the output of each node for each channel-commodity pair have been used; a mechanism for loading the output queues on different channels has been defined introducing the notion of virtual links.

The algorithm has been presented for a general communication and interference scenario. In order to test the behavior of the full algorithm, an instance of the problem, based on a simplified communication and interference model, has been simulated using a greedy centralized scheduler.

The network performance has been evaluated as a function of the number of channels, interfaces and traffic flows. Three different schedulers have been considered. The results are consistent with previous theoretical findings, and confirm the goodness of the approach. On the other hand, the specific features of our algorithm could give some insight on practical implementations in a distributed setting.

3.10 Acknowledgments

The work presented in this chapter is a joint work with Prof. Nitin Vaidya, Wireless Networking Group at University of Illinois at Urbana Chamapaign.

Chapter 4

Optimization in IEEE 802.11 WLAN

This chapter addresses the problem of optimizing the performance of an IEEE 802.11 Wireless LAN. The IEEE Standard specifies the PHY, MAC and LL mechanisms, but some optimization aspects are left open for the developers which are challenged to find optimized solutions according to their needs.

The setting of LL, MAC and PHY parameters, such as physical transmission rate, maximum retransmission number of MAC frames, frame lengths, etc. has a dramatic impact on the performance of wireless links. We are interested in adapting such parameters by means of practical algorithms based on solid analytical modeling of the standard MAC and PHY behavior.

Based on the current network status, and using mathematical models for the MAC, PHY and LL mechanism we are able to predict the link performance in terms of packet loss, throughput, delay under different settings. In particular, the estimation of the current network status turns out to be a fundamental task in this operation. By linking this model with different optimization objectives we are able to define several optimization algorithms spanning different functional layers.

In particular, rate adaptation algorithms which are adaptive to the propagation and interference conditions are designed.

Moreover, by suitably merging the model for the link performance with a model for the quality of voice connections, we also optimized the performance of VoIP over WLAN communications.

4.1 Introduction

In wireless systems, such as WLANs, the propagation environment and the interference scenario change over time and space due to factors such as mobility, propagation dynamics and traffic variations.

To cope with this challenging environment, many wireless interfaces offer the possibility of dynamically tuning some system parameters in order to adapt to the environmental variations. The IEEE 802.11 specifications, in particular, define a plurality of PHY modes which can be used for the transmission of data frames. Each PHY mode uses a particular modulation and channel coding scheme and, consequently, offers different performance in terms of transmission duration, overhead, and robustness against reception noise and interference. Moreover some other parameters are tunable, such as the maximum number of retransmission at the MAC layer and the frame size. Many other parameters which are defined as non tunable in former standard versions have been recognized as of basic importance for determining the network performance and thus introduced as tunable parameters in the 802.11e version. MAC specifications for this recent standard allow to tune the backoff window length and the interframe timing, by also introducing different quality of service classes.

Thus, some degrees of freedom are available to the designers which can tune the network setting in order to achieve a broad range of optimization goals. By linking these parameters settings to the corresponding network performance and defining suitable optimization objectives, the design of cross layer optimization problems is made possible. Then, optimization algorithms have to be designed in order to actually achieve the solution with reasonable complexity.

In particular we are interested in algorithms which act on the settings of a single IEEE 802.11 link to adapt its behavior to the current interference and propagation conditions. Two entities are thus defined in our formalization, i.e. the wireless link to be optimized and the status of the network. We define an analytical model for the link performance based on a mathematical model for the LL, MAC and PHY mechanism which accounts for the status of the working environment.

Based on the analytical modeling, the triplet $\langle \text{SNR}, P_{coll}, \xi \rangle$ is identified as a sufficient description for the working environment. This triplet will be referred to as Medium Status in the following. In this triplet SNR is the experienced Signal to Noise Ratio, P_{coll} is the collision probability and ξ is the average tick period, defined as the time between two successive decrements of the backoff counter. Please note how the Medium Status accounts for propagation characteristics of the link, through SNR, interference coming from concurrent transmitters, through P_{coll}, and medium congestion, through ξ, which turns out to be proportional to the duration of channel occupancy from concurrent nodes.

Thus, the proposed Medium Status definition allows, for instance, for the explicit distinction between packet losses due to channel impairments and those due to medium congestion and interference. This fundamental distinction will be accounted in the analytical model for the link performance and allows for the definition of optimization algorithms which are able to effectively adapt to complex working environments where channel condition and medium congestion are jointly addressed.

Approximations for practical implementation will be discussed later on in this chapter.

In order to make it possible to implement our scheme on real devices, we provide a method for estimating the Medium Status based on information which is commonly available on commercial devices in the form of 802.11 Management Information Base (MIB) counters,[1] with the exception of an event counter which we propose to add and which could be easily implemented by device manufacturers. The correctness of the estimation in a broad range of operating condition is verified through detailed simulations.

Based on the knowledge of the Medium Status, the analytical model is able to predict all the major statistics needed to characterize the performance of a link, such as packet dropping probability, delay, throughput etc..

This framework is used to define multiple optimization problems with different objectives. The first application presented in this chapter refers to a Rate Adaptation (RA) algorithm.

In recent years, there has been a significant amount of research on this topic. In particular, the case of a single sender/receiver pair has been deeply investigated; for this type of scenario, in which packet losses are due to channel impairment only, an optimal RA strategy has been proposed [62], as well as several practical RA algorithms [64, 63, 65, 66, 67].

However, in typical 802.11 scenarios, multiple users are contending for the medium. Therefore, in addition to channel-related packet losses, also MAC collisions and variable medium access time have a significant impact on performance, and in practice make the above mentioned schemes sub-optimal and, in some cases, very inefficient. Some recently proposed practical RA schemes [69, 68] address the problem of collision-related packet losses; however, to the best of our knowledge, no previous work formulated an optimal RA policy for multi-user 802.11 scenarios.

Thanks to the previously described optimization framework, we try to fill this gap by proposing a novel RA algorithm, named Goodput Optimal Rate Adaptation (GORA) which uses Medium Status estimation together with an analytical model of the goodput performance for 802.11. GORA selects the PHY mode that, according to the outcome of the analytical model, yields the best throughput for the given estimated Medium Status.

We tested GORA by using both ideal perfect Medium Status knowledge and the actual outcome of the proposed Medium Status Estimator.

Simulations results by using ideal Medium Status estimation show that GORA mechanism always outperforms other RA algorithms (also provided with perfect information), thus showing how it can be used as a new benchmark for practical rate adaptation schemes, especially in scenarios with interfering transmissions.

Furthermore, the practical version of GORA, which uses the actual outcome of the proposed Medium Status estimator, achieves performance that is close to that with ideal Medium Status estimation, thus confirming that the RA framework here proposed is very effective in a number of different scenarios.

In the second application, the outcome of the link performance model is used to compute the expected quality of a voice connection by means of a mathematical model which empirically maps transmission performance to perceptual voice quality.

[1] The formal specification of the 802.11 MIB is Annex D of the 802.11 specification.

The considered utility function is the one proposed in [70] for Voice over IP, which is inspired by the E-Model defined in ITU-T recommendations G.107 [102] and G.113 [103]. Such a utility function returns a numerical rating of the voice quality that can be mapped to the Mean–Opinion–Score (MOS), which we consider as the objective function of our optimization problem. This mathematical model allows to map delay and packet loss ratio experienced on the link to a 1-to-5 MOS index representing the perceptual quality of the voice connection.

The resulting optimization framework is used to jointly tune the physical transmission rate and the maximum number of allowed retransmissions for a single packet at the MAC level, in order to maximize the MOS index.

Whether the E-model is an effective way of representing voice quality is not discussed in this chapter, as we are more interested in showing the potential benefit achievable by the proposed cross layer optimization framework.

We use an enhanced version of the NS2 simulator to test the performance of the proposed optimization algorithms. The simulator is built by using a recently developed modular architecture [104] and embeds an enhanced propagation and interference model, together with the multirate capability of IEEE 802.11b/g networks.

The rest of the chapter is organized as follows. In Section 4.2 we introduce the reference framework and the basic constituent blocks. In Section 4.3 we define the Medium Status discussing how it can be estimated on real devices. The model for the link performance is introduced in Section 4.4. In Section 4.5 an instance for the optimization block is presented, realizing the GORA rate adapter. Simulation results and comparisons with previous results are also presented. In Section 4.6 a second instance for the optimization block is presented, optimizing the quality of a VoIP connection, together with simulations results. Finally, in section 4.7, the conclusions are drawn.

4.2 Optimization framework

The proposed framework can be described by four blocks:

- *MSE: Medium Status Estimation.* This block provides an estimation of the medium status.
- *WLM: Wireless Link Model.* This block provides an estimation of the end–to–end performance that is expected, given the estimated medium status and the working parameter setting.
- *QEB: Quality Evaluation Block.* This block computes the objective function Q
- *OPT: Optimization.* This block determines the optimal setting for the tunable parameters in order to optimize the value of Q.

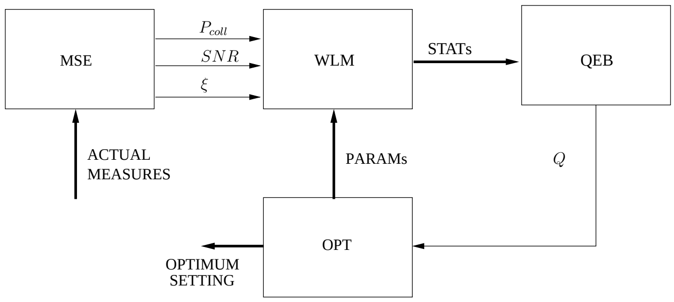

Figure 4.1: Optimization Framework

Note that in case an analytical solution for the optimization problem were available, the proposed framework could be simplified by appropriately designing a single functional block, whose inputs are the network status and the corresponding output is the optimal parameter settings for a given utility.

Unfortunately, the complexity of the MAC mechanisms do not allow for such a complete analytical description and thus the proposed architecture has been designed in order to allow the description of an exhaustive search for the optimal solution.

Moreover the proposed architecture is designed to be modular enough to host different utility functions, medium estimators and optimization mechanisms, which can be independently interchanged. In this sense the proposed architecture represents a new architecture for a functional layering.

Thus, according to the proposed framework, the optimization process consists in iterating the computation of the wireless link performance and the related utility for different choices of the tunable parameters, until the optimal setting is found. In case some analytical properties relating the medium status with the optimal setting are known (monotonicity, physical constraint) the search process can be improved for a better efficiency.

A basic assumption used throughout this chapter is that the network status does not depend on the specific setting of the link being optimized. This clearly represents a simplified assumption.

4.3 Medium Status estimation

This block provides a sufficient description of the current network status which allows the WLM block to correctly predict the link performance over a broad range of values for the tunable parameters.

Based on the analytical model for the link performance, we identified some key metrics which represent a sufficient medium status description for our purposes. The Medium Status is thus defined as the triplet $\langle \text{SNR}, P_{coll}, \xi \rangle$, where SNR is the Signal to Noise Ratio at the receiver, P_{coll} is the Collision Probability experienced by the mobile station (STA) and ξ is the average *tick period*, defined as the time between two successive decrements of the backoff counter.[2] This Medium Status definition allows to take into account all PHY and MAC layer aspects in a 802.11 STA.

The SNR accounts for propagation characteristics of the link as it only depends on the transmitted power (which we consider to be fixed), the channel gain and the thermal noise at the receiver. Note that, once the relation between SNR and packet loss is available, we can also define the quantity P_{err} as the packet loss depending on the experienced SNR.

P_{coll} accounts for the fraction of transmitted packets which have been erroneously received because of interference. Clearly this quantity is not well defined as the packet loss in general depends on the SNIR (Signal to Noise and Interference Ratio) which jointly accounts for interference, channel propagation and thermal noise. We conclude that the two events are not mutually exclusive, and that both contribute to the packet loss event, as represented in Figure 4.2.

Thus we resort to a simplifying definition which is suitable only for particular, but nonetheless frequent scenarios. In particular we assume that interfering nodes are close enough to the receiver so as to corrupt any concurrent transmission. This collision model allows us to connect part of our investigation to the interference model used in the well known performance study in [105].

We define P_{loss} as the total fraction of lost packets. Since P_{loss} accounts for the contribution of both P_{coll} and P_{err} we also have

$$P_{\text{loss}} = P_{\text{coll}} + P_{\text{err}} - P_{\text{coll}} P_{\text{err}}. \tag{4.1}$$

Based on the MAC mechanism, which requires the backoff counter to freeze whenever a transmission is sensed on the medium, the ξ measure turns out to be proportional to the time the medium is occupied by an interfering transmission. Thus, this quantity represents the congestion on the medium.

Such a model for the medium status allows for a differentiation between packet losses due to interference (by means of P_{coll}) and packet losses due to channel impairments (by means of P_{err}). This distinction is fundamental in order to allow for a correct adaptation of the network setting, discerning bad medium status due to propagation conditions from that due to a high number of concurrent transmission. Clearly, different settings are optimal in the two cases.

As we are interested in algorithms which are close to practical implementation, a key point in our proposal is the possibility of achieving a good estimation of the aforementioned status variables by using information available in actual devices with a reasonable complexity.

With this respect we propose a Medium Status Estimation method which is based on the use of some measurements available at the MAC layer. These measurements, and the

[2] In [105] this is called *slot period*. We prefer not to use the term *slot* in order to avoid possible confusion with the 802.11 PHY slot time – the difference is that the tick period can be much longer than the slot time due to the backoff freeze procedure.

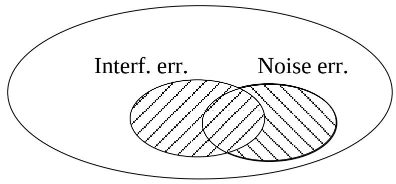

Figure 4.2: Packet loss model

notation which will be used for them throughout this paper, are:

- t_s: the number of successfully transmitted unicast MSDUs, i.e., the number of transmitted data frames for which an ACK was received;
- t_f: the number of transmitted data frames for which an ACK was not received;
- r_s: the number of successfully received data frames;
- r_f: the number of received frames for which the checksum failed;
- s_i: the number of idle time slots, except the ones preceded by a transmission.

All these measurements can be obtained directly or indirectly by some of the counters available within the 802.11 Management Information Block (MIB).[3] The only exception is the idle time slots counter s_i, which is not listed among the counters in the MIB; we note, however, that its implementation would be rather straightforward, and therefore our proposal still maintains a high degree of implementability in real devices.

We suppose all above mentioned counters to refer to the events occurred in a time window of given duration D.

For convenience we define the following variables:

- $t_a = t_s + t_f$ the total number of frame transmission attempts performed;
- $s_b = r_f + r_s$ the number of busy time slots, which corresponds to the number of transmission attempts made by other transmitter while the link under consideration is not transmitting;
- t_c number of times a transmission attempt by the link under consideration fails due to collision;
- t_n the number of transmission attempts which are not affected by collision (though they might be affected by channel errors);

[3] We refer to the `dot11Counters` described in the the IEEE 802.11 standard, Annex D [106]. We note that, in order to derive the measurements we need from the MIB counters, some processing is required, since some counters also include control and management frames, while the measurements we use are supposed to count only for data frames.

In the following we will describe how Medium Status Estimation is performed.

Following the same approach proposed in [105], we assume that both the transmission and the collision probability are stationary, i.e., independent of the particular slot considered. With this assumption, the collision probability for a transmission by the link under consideration equals the probability that a randomly selected time slot is occupied by a transmission from another node. Please note that, due to the MAC mechanism, which requires the backoff counter to freeze while the medium is sensed busy, we are not interested in the time slots elapsed during the freezing period. We are only interested on the time slots where a transmission start is possible. In this sense, the transmission or the reception of a packet is only counted as a single time slot, since a collision could happen only on the first slot.

We now compute the needed events based on the available counters measurements.

The number t_f of failed transmission attempts can be expressed as the P_{loss} fraction of all attempts

$$t_f = t_a P_{\text{loss}} \tag{4.2}$$

and thus, using (4.1) we also get

$$t_f = t_a P_{\text{coll}} + t_a(1 - P_{\text{coll}})P_{\text{err}} \tag{4.3}$$

The number t_c of collided transmission attempts can be expressed as the P_{coll} fraction of all attempts

$$t_c = t_a P_{\text{coll}} \tag{4.4}$$

The number of non collided transmissions can be expressed as

$$s_n = t_a - t_c \tag{4.5}$$

Based on the previous definitions, P_{coll} can be evaluated with the following formula:

$$P_{\text{coll}} = \frac{s_b + t_c}{s_b + s_i + t_c + s_{nc}} \tag{4.6}$$

where, according to its definition

$$s_b = r_f + r_s \tag{4.7}$$

The numerator of (4.6) represents the number of busy slots, and the denominator represents the total number of slots.

Substituting (4.5), (4.7) and (4.4) into (4.6) we get

$$P_{\text{coll}} = \frac{r_f + r_s + t_a P_{\text{coll}}}{r_f + r_s + s_i + t_a} \tag{4.8}$$

which leads to the following estimator for the collision probability:

$$P_{\text{coll}} = \frac{r_f + r_s}{r_f + r_s + s_i}. \tag{4.9}$$

Please note that the resulting P_{coll} does not depend on the counters related to the transmissions of the STA under consideration, but only on the events caused by all the other STAs. This is the main difference with the estimator presented in [107], which allows our estimator to be effective also in the presence of packet losses due to channel impairment.

Since P_{loss} can be obtained as the ratio between the erroneous packets and the total transmitted packets:

$$P_{\text{loss}} = \frac{t_f}{t_f + t_s}.$$

With minor algebraical manipulation the packet loss ratio due to channel impairment P_{err} can be derived as

$$P_{\text{err}} = \frac{t_f - (t_f + t_s)P_{\text{coll}}}{(t_f + t_s)(1 - P_{\text{coll}})}. \tag{4.10}$$

We suppose that P_{err} is univocally determined by a known function of the PHY mode being used, the packet size and the SNR (without considering the interference effect) seen by the receiver. The considered function is represented in Figure 4.3. We suppose SNR is constant for the whole packet transmission duration. Then we determine SNR by inverting the SNR versus P_{err} relation for the rate being used. Clearly this practice requires that the same PHY mode was used for the whole observation period D.

In order to simplify the analytical study, we assume that ACKs are always correctly received, which is a reasonable assumption in a scenario where all nodes are in the same collision domain and the channel is symmetric.

The average tick period ξ can be estimated by dividing the total time the channel was sensed busy by the considered station over the total number of backoff tick periods in the observation window:

$$\xi = \frac{D - t_f T_f - t_s T_s - t_a \sigma}{s_i + r_s + r_f}, \tag{4.11}$$

where σ is the PHY slot time, and T_f and T_s are the duration for respectively a failed and successful frame transmission by the STA being considered. The complete derivation is not reported for conciseness. We note that T_s is determined only by the payload size and the PHY mode being used, whereas T_f is a random variable which depends also, in the case of a collision, on the duration of the transmissions performed by other users. In detail, we have

$$T_f = \frac{P_{\text{coll}}}{P_{\text{loss}}} T_c + \frac{1 - P_{\text{coll}}}{P_{\text{loss}}} T_e, \tag{4.12}$$

where T_c is the duration of a collided transmission and T_e is the duration of a frame loss due to channel errors. These values can be approximated as

$$T_c = \max(T_B, T_s) \tag{4.13}$$

$$T_e = T_s - (T_{ACK} + DIFS + SIFS) + EIFS \tag{4.14}$$

with T_B the duration of the colliding packet.

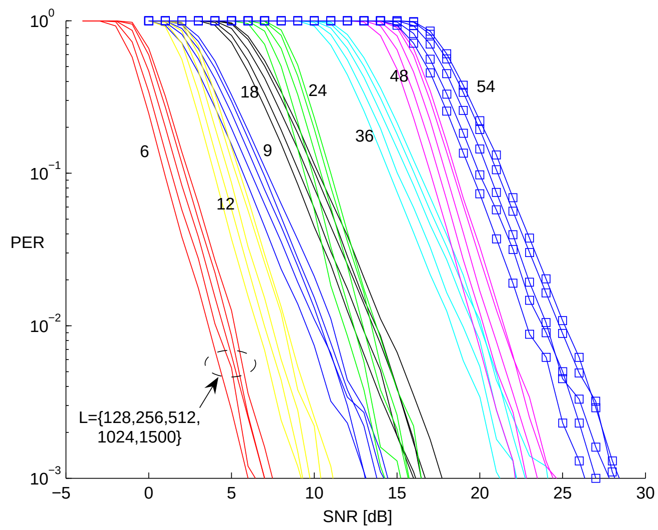

Figure 4.3: Packet loss ratio versus SNR for a IEEE 802.11g connection.

4.3.1 Estimations validation

In this section, the proposed estimation technique for P_{coll} and P_{err} is validated.

In Figure 4.4 P_{coll} is compared with the analytical value derived in [105]. In this scenario no errors due to channel impairment are present so that $P_{coll} = P_{loss}$. Results shows that our estimation is very close to the analytical model in [105] when in the same conditions.

In Figure 4.5 we report the results for a scenario where the test node is placed at an increasing distance from the AP, and a fixed number of interfering nodes are placed close to the AP. In this case errors due to channel impairment are present, thus the model in [105] can not be directly applied. P_{coll} and P_{err}, estimates according to formulae (4.9) and (4.10), are compared with the actual values measured from the trace files. As can be seen, estimated values closely match the actual ones.

In Figure 4.6 P_{err} and P_{coll} are shown as a function of the number of interfering nodes. As can be seen, as the number of interfering nodes increases P_{coll} increases. P_{err} is almost constant as it should be, representing the packet loss related to the SNR value, which is constant in this scenario.

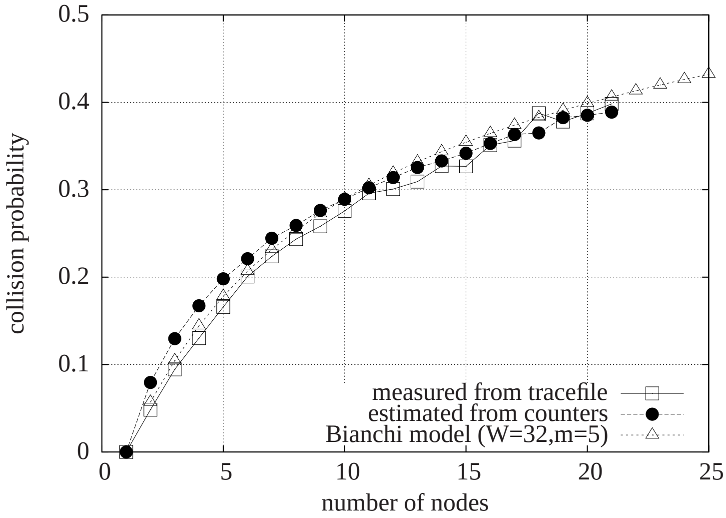

Figure 4.4: Accuracy of the P_{coll} estimator in an error-free scenario

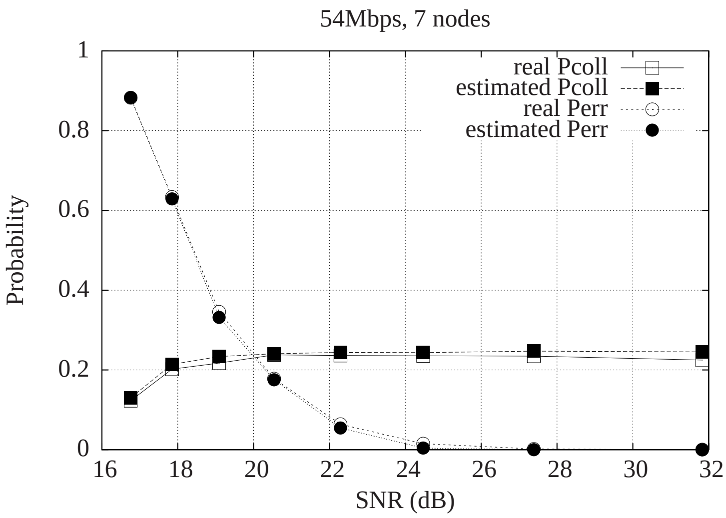

Figure 4.5: Accuracy of the P_{coll} estimator as a function of the distance between the test node and the AP, in the case of 6 concurrent transmissions

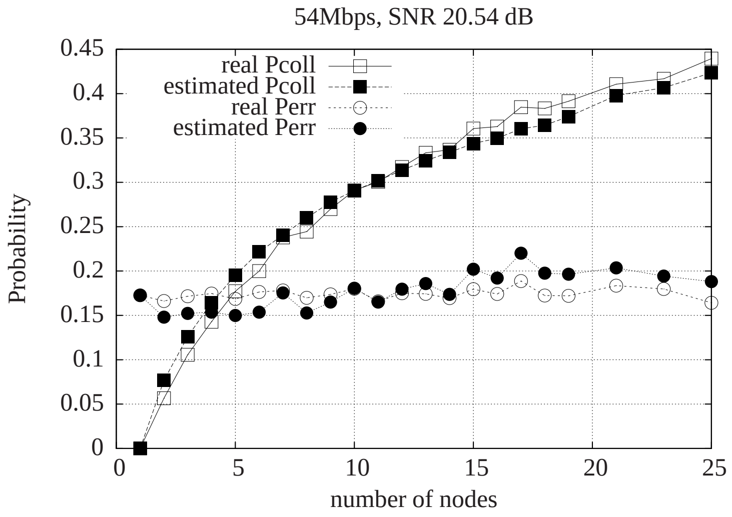

Figure 4.6: Actual and estimated P_{coll} and P_{err} as a function of the number of interfering nodes for $SNR = 20$dB.

We obtained similar accuracy in a wide range of values for the SNR and the number of interfering nodes, and we can therefore conclude that the proposed estimator is able to provide an accurate and distinct information on the propagation environment and congestion level.

4.4 Link performance model

In this section the performance of the link transmission is derived. We focus on a Constant Bit Rate traffic source and assume a saturation condition, i.e. the queue at the source is never empty. Under these assumptions we derive the statistical description of the packet dropping and delay experienced by packets, which will be used by the optimization algorithms presented in the next sections.

A packet is dropped when it is embedded in a MAC Packet data Unit (MPDU) that is not successfully delivered after r_{max} transmission attempts, where r_{max} is the max retry limit parameter. Denoting by P_{loss} the packet failure rate at the MAC layer, we have

$$P_{\text{drop}} = P_{\text{loss}}{}^{r_{\text{max}}}.$$

For deriving the delay statistics, as a first step we need to introduce the following notations.

- ξ_j: duration of the j–th tick period[4] seen by the tagged mobile station during backoff.

[4] A tick period is defined as the time period between two consecutive decrements of the backoff counters of backlogged stations (see [108]).

- W_0: minimum contention window size.
- W_h: backoff window size at the h-th transmission attempt: $W_h = 2^{\min(h,m)} W_0$, where m is the maximum number of backoff stages defined by the standard.
- y: MAC layer service time, i.e., time taken by the MAC entities to get rid of the head of the queue MPDU, either successfully delivering the packet to the destination or dropping it after r_{max} failed transmission attempts. y' denote the service time for successfully delivered packets only.
- w: queuing time, i.e., time spent by a packet in the MAC sender queue.
- s: system time, i.e., time elapsed from the packet generation to its delivery. It accounts for both successfully delivered packets and those dropped after r_{max} failed transmission attempts. s' accounts only for successfully delivered packets.
- m_x, M_x, $\mathcal{M}_x$: first, second and third order moment of the (generic) random variable x, given by the expectation of x, x^2 and x^3, respectively. Also, σ_x^2 denotes the variance of the r.v. x.

To analyze the delay statistics of the delivery of packets over the wireless link, we model the 802.11 MAC layer as a queue–server system, with infinite queue space, customer arrival rate λ and stochastic service time y. According to this model, each customer corresponds to a MPDU, while the service time y corresponds to the time taken by the MAC entity to process a MPDU, that is the time elapsed since the MPDU is fetched from the head of the MAC queue until either the relative ACK frame is correctly received or the frame itself is discarded due to exceeded retransmission limit r_{max}. Therefore, the service time, which is the overall time spent by the packet in the MAC layer, is given by the sum of the queuing delay w and the service delay y:

$$s = w + y \ .$$

The queuing delay w, in turn, depends on the service time y and the arrival rate λ.

The derivation of the statistics of y is long and cumbersome. Here the problem is formulated and only the first order statistic is derived. Details on the complete derivation are reported in [109].

During the service of a MPDU, the MAC entities alternate between two different phases, namely backoff and transmission. The backoff procedure, in general, is performed before any transmission attempt. The stage of the backoff procedure is set to zero any time a MPDU is served (either successfully delivered or dropped) and, then, incremented by one at each retransmission attempt. At each backoff stage, the MAC entity picks a random integer value in the set within the backoff window associated to that stage, and countdowns by one every tick period. Therefore, denoting by $B(h)$ the time spent in the h–th backoff stage, we have

$$B(h) = \sum_{r=0}^{W_h-1} \chi_b(r) \sum_{j=1}^{r} \xi_j \; ; \tag{4.15}$$

where $\chi_b(r)$ equals 1 or 0 depending on whether or not the backoff value picked when the h-th backoff stage was entered is equal to r, given that the backoff window was $\{0, 1, \ldots, W_h - 1\}$. As discussed in Section 4.3 we assume to have an estimation for the m_ξ value.

By taking the expectation of (4.15) we get

$$m_{B(h)} = \sum_{r=0}^{W_h-1} r \frac{m_\xi}{W_h} = \frac{W_h - 1}{2} m_\xi \; ; \tag{4.16}$$

where we have replaced $E[\chi_b(r)]$ with $1/W_h$, given that the backoff value is uniformly chosen in a window of W_h integer values.

We can now express the service time y as function of the time spent in transmission and backoff. Let $\chi_s(i)$ be equal 1 or 0 whether or not the packet is successfully delivered after i consecutive failures. Similarly, let χ_d be 1 if the transmission has failed for r_{max} successive attempts (so that the packet is dropped for reached max retry limit) and 0 otherwise. Then, we have

$$y = \sum_{i=0}^{r_{\text{max}}-1} y_s(i)\chi_s(i) + y_d \chi_d \; ; \tag{4.17}$$

where

$$y_s(i) = \sum_{h=0}^{i} B(h) + iT_F + T_s \; ; \quad y_d = \sum_{h=0}^{r_{\text{max}}-1} B(h) + r_{\text{max}} T_F \; .$$

Keeping into consideration that the expectation of the indicator functions $\chi_s(i)$ and χ_d is given by

$$E\left[\chi_s(i)\right] = P_{\text{loss}}^{i}(1 - P_{\text{loss}}) \; ; \quad E\left[\chi_d\right] = P_{\text{loss}}^{r_{\text{max}}} \; ;$$

then, the average service time can be expressed as

$$\begin{aligned} m_y &= \sum_{i=0}^{r_{\text{max}}-1} P_{\text{loss}}^{i}(1 - P_{\text{loss}}) \left(\sum_{h=0}^{i} E[B(h)] + iT_F + T_s \right) + \\ &\quad + P_{\text{loss}}^{r_{\text{max}}} \left(\sum_{h=0}^{r_{\text{max}}-1} E[B(h)] + r_{\text{max}} T_F \right) \\ &= \sum_{i=0}^{r_{\text{max}}-1} P_{\text{loss}}^{i}(1 - P_{\text{loss}}) \left(\sum_{h=0}^{i} \frac{W_h - 1}{2} m_\xi + iT_F + T_s \right) + \\ &\quad + P_{\text{loss}}^{r_{\text{max}}} \left(\sum_{h=0}^{r_{\text{max}}-1} \frac{W_h - 1}{2} m_\xi + r_{\text{max}} T_F \right) \; . \end{aligned} \tag{4.18}$$

Notice that m_y also includes the service time of dropped packets. The average service time for successfully delivered MPDU is, instead, given by

$$m_{y'} \quad = \quad \sum_{i=0}^{r_{\max}-1} \frac{P_{\text{loss}}^i (1 - P_{\text{loss}})}{1 - P_{\text{loss}}^{r_{\max}}} \left(\sum_{h=0}^{i} \frac{W_h - 1}{2} m_\xi + iT_F + T_s \right) . \tag{4.19}$$

This results will be used in Section 4.5.

For the optimization objective presented in Section 4.6 we need further statistical measures. In particular we will need the first and second order moments of the system delay. The derivation of this quantity is long and cumbersome and requires the computation of additional moment of y as well. Here its derivation is sketched and the final result is reported. The complete derivation can be found in [109].

By modeling the arrival and departure process by means of a statistical multiplexer we can derive the following approximations for the first and second order moments of the system time s.

$$m_s = m_y + \frac{\lambda \sigma_y^2}{2(1-\rho)} . \tag{4.20}$$

$$M_s = M_y(1-\rho) + \frac{\lambda \mathcal{M}_y}{3} + m_x(\sigma_y^2 + M_y) + M_x m_y^2 . \tag{4.21}$$

We have thus derived all the necessary statistics that will be used by the proposed optimization algorithms presented in the next sections. Based on the detailed model presented in [109], further statistical performance can be computed, but they are not reported in this thesis.

4.5 Optimization block: Rate adaptation

4.5.1 Algorithm description

The proposed rate adaptation algorithm uses the outcome of Medium Status estimation $\langle \text{SNR}, P_{\text{coll}}, \xi \rangle$ to compute the expected goodput for all the possible PHY modes. The average goodput G of the system can be expressed as

$$G = \frac{L}{m_y} (1 - (P_{\text{loss}})^{r_{\max}}) . \tag{4.22}$$

where L is the payload length. The goodput is computed for different transmission rates and PHY modes; the one which achieves the highest goodput is selected and used to transmit the data frame.

Note that changing the PHY modes impacts the goodput in two ways. The transmission time increases as the rate decreases and the loss probability decreases as the rate decreases.

Depending on the P_{err} and P_{coll} values, the two effect have a different impact on the goodput. Thus, given a fixed P_{loss} different rates turns out to be optimal depending on the P_{err} and P_{coll} values.

The optimization is performed periodically every T_{opt} seconds. As the Medium Status is estimated at run time by collecting MAC counters statistics, the duration of T_{opt} is related to D. We note that both values should be chosen taking into account the desired tradeoff between an accurate estimation and a fast PHY mode adaptation.

4.5.2 Performance simulation

In this section we report the performance evaluation of the proposed rate adaptation mechanism.

Simulations have been performed by using an enhanced version of the NS2 simulator [104]. All the specific parameters for the network setting are compliant with the IEEE 802.11g standard. In particular a maximum retry limit of 7 has been used. The propagation and interference model used in the simulations is based on a Gaussian approximation for the interference as widely assumed in literature. The channel model is simply determined by the two ray ground model accounting only for the path loss component. Fast channel gain fluctuation are not considered.

The model for the PHY layer transmission performance is described in terms of the packet loss probability as a function of the PHY mode, SNIR, and packet length. This characterization has been computed offline using a dedicated PHY layer simulator accounting for the standard specification of the OFDM modulation and coding implementation.

The considered scenario refers to an infrastructured network where a test STA is connected to the access point (AP) and is provided with the parameter estimation and rate adaptation algorithms. Other STAs are connected to the same access point in order to simulate an interfered scenario. Such interfering STAs are not provided with the rate adaptation algorithms.

Only uplink connections are simulated. All the traffic sources are CBR over UDP, with a packet length of 1500 bytes, and a generation rate such that all the connections are in a saturated conditions.

We test the proposed GORA algorithm both in the case an exact SNR value is used and in the case the SNR value is provided by the proposed estimator.

The algorithms are compared with ARF, MBLAS (which assumes to have perfect SNR knowledge) and a modified version of MBLAS which uses the RTS-CTS mechanism to receive information on the SNR at the receiver.

In Figure 4.7 we report the results for a scenario with no interferers. In this case, MBLAS has been shown to achieve optimum performance. The proposed GORA algorithm, both with exact and estimated SNR achieves the same optimum throughput. These algorithms outperform ARF and the MBLAS version which suffers from the RTS-CTS overhead.

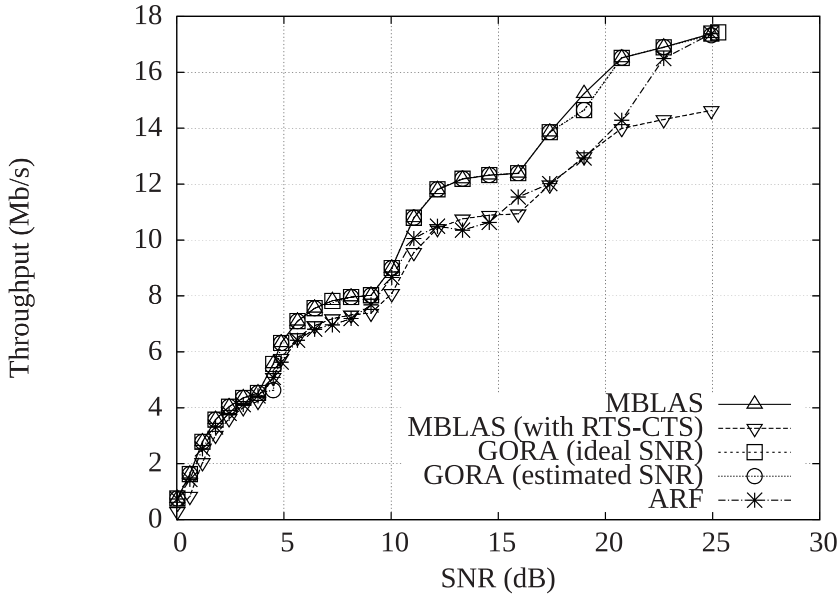

Figure 4.7: Throughput of rate adaptation algorithms, as a function of the distance between the test node and the access point, in case of no interference

In Figure 4.8 results for a scenario with 9 interferers is presented. In this case, MBLAS turns out to be suboptimal. This is due to the fact that MBLAS does not account for the variations in the medium access time and thus neglects the increased transmission delay in case of congested medium. Instead, the proposed GORA mechanism is able to adapt the PHY mode selection to the medium status and in the case an exact SNR knowledge is considered, it always achieves the highest throughput. When using the estimated SNR value, GORA still shows good performance. In particular we stress that GORA is a standard compliant mechanism and can be implemented in actual network cards. In this regard we are interested in the comparison with ARF, which is highly deployed in practice, and with MBLAS with RTS-CTS, which is close to practical (even if not standard compliant) implementation. Both algorithms are outperformed by the proposed mechanism in a wide range of working conditions.

In Figure 4.9 results for a scenario with a fixed SNR and a variable number of interferers is presented. In this particular setting, GORA both with exact and with estimated SNR outperforms the other rate adaptation algorithms.

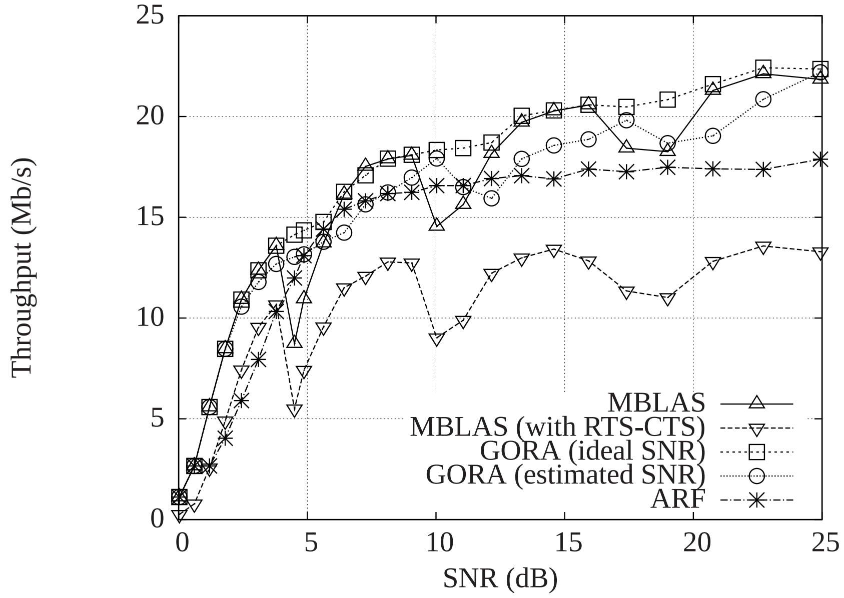

Figure 4.8: Throughput of rate adaptation algorithms, as a function of the distance between the test node and the access point, in case of 9 interfering nodes

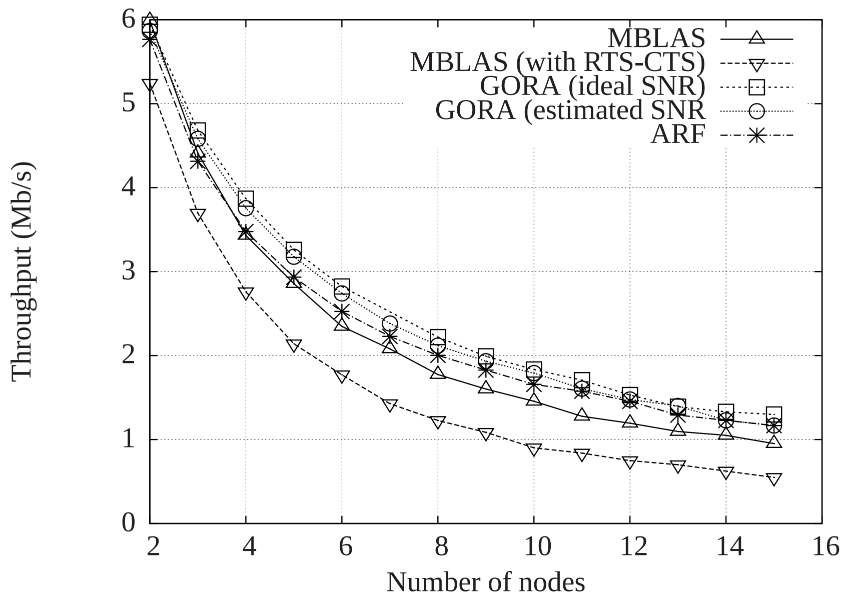

Figure 4.9: Throughput of rate adaptation algorithms, as a function of the number of interfering nodes, with a fixed distance of 36.91m between the test node and the access point

4.6 Optimization block: Voice quality enhancement

Following the footprint of [70] and using the quality evaluation function defined in 2.4.1, in this section it is defined an optimization block that can fit in the previously defined framework and can be used to improve the quality of VoIP communications.

In this section, for illustrative purposes, we assume a simple, but rather common, playout buffer model. More specifically, we assume that the first packet of each talk spurt is held in the playout buffer for a fixed amount of time, δ_{buff}, before being released to the decoder. After this initial delay, voice packets are regularly fetched from the playout and passed to the coder. We assume that the buffer size is large enough to avoid overrunning. Then, packet loss can occur in case of underruns only.

Now, let us consider a talkspurt of M voice packets, where M is large. Furthermore, let s_i be the system delay of the i–th packet, i.e., the time elapsed from the epoch the packet was generated till it was received by to the playout buffer. Assuming that the wired part of the connection does not introduce significant delay then, the loss due to playout buffer underrun can be generally expressed by using a conservative Chebyshev bound:

$$\delta_{buff}^2 P_{buff} \leq 2\sigma_s^2 \ .$$

Therefore, P_{buff} can be approximated by

$$P_{buff} = \min\left(\frac{2\sigma_s^2}{\delta_{buff}^2}, 1\right) \ . \tag{4.23}$$

This is a quite rough approximation due to the complex model. Please notice that the jitter in the packet arrival is translated to an increased packet loss an in creased delay, due to the presence of the playout buffer.

4.6.1 Optimization algorithm

The MOS metric described in Section2.4.1, defined by Equation (2.11) and the related Equation (2.12), is used as the optimization objective to improve the quality of the voice connection. As already discussed, such metric is a function of the codec type, the packet delivery ratio, and the mouth-to-ear delay. Thus eventually the MOS can be expressed as a function of the packet loss ratio P_{loss}, the packet dropping ratio at the buffer P_{buff} (which depends on the jitter, measured by σ) and the delay s. Thus, by exploiting the relation between the MOS and the link performance metrics, and by using the mathematical model which relates the link performance metrics to the MAC tunable parameters setting, an optimization problem can be defined which acts on such MAC parameters in order to optimize the MOS.

In this thesis we are not interested in evaluating the effectiveness of the MOS metric in describing the quality of a voice connection. We can generally assume the actual voice quality to be a monotonic increasing function of the MOS so that we can use the MOS as a metric to compare different solution in a relative way.

In particular, in order to prove the importance of a cross layer optimization in this context, two optimization algorithms are tested. The fist one only performs rate adaptation, thus finding the most suitable rate for maximizing the MOS. The second one instead jointly maximizes the transmission rate and the maximum retry limit r_{max}, by pursuing the same optimization objective.

Since the parameters take value on finite and rather small discrete sets, the easiest way to find the optimal setting is to perform an exhaustive search over the value space of the tunable parameters. Stability constraints as well as monotonic dependencies of voice quality to some parameters could be exploited to reduce the search space and, hence, the computational load. Notice that the optimization process carried out by the optimization block can be performed off–line over a given subset of the admissible space values for the state variables returned by NSE block, namely P_{coll} and SNR. Then, the corresponding optimal parameter setting can be stored in a memory, in order to avoid the need of implementing the optimization framework on the terminal. However, more advanced and sophisticated strategies might be envisioned, also depending on the objective function considered.

4.6.2 Optimization validation

The proposed optimization was tested by using an enhanced version of NS2. The simulation scenario is the same described in Section 4.5.2. In particular a G.711 voice codec is considered.

In order to test different propagation conditions the VoIP link has been tested by placing the mobile station at an increasing distance from the AP. In order to test different congestion conditions, the test has been repeated with an increasing number of interfering nodes.

In Figure 4.10 results for the algorithm performing only rate adaptation are reported. In this case the maximum retry limit is set to $r_{max} = 7$. As can be seen the MOS index reaches satisfactory values only for the scenarios where 1 or 3 interfering nodes are present. On the contrary, in Figure 4.11 results for the algorithm performing both rate adaptation and maximum retry limit adaptation are reported. The MOS in this case is significantly higher than in the previous results set. We thus conclude that the r_{max} parameter plays a fundamental role in the optimization of voice connections. The retry limit acts as an important parameter in determining both the reliability and the latency of the connection. In particular, packet retransmissions in heavily congested scenarios experience high channel access delay which in turn causes high jitter and high packet dropping rate at the playout buffer, since the delay of a given packet affects also the transmission of the successive ones. In this scenario, an early discard of delayed packets experiencing retransmissions, by imposing a smaller retransmission limit, can be beneficial for the delay of successive packets, while causing a smaller increase in the packet loss ratio.

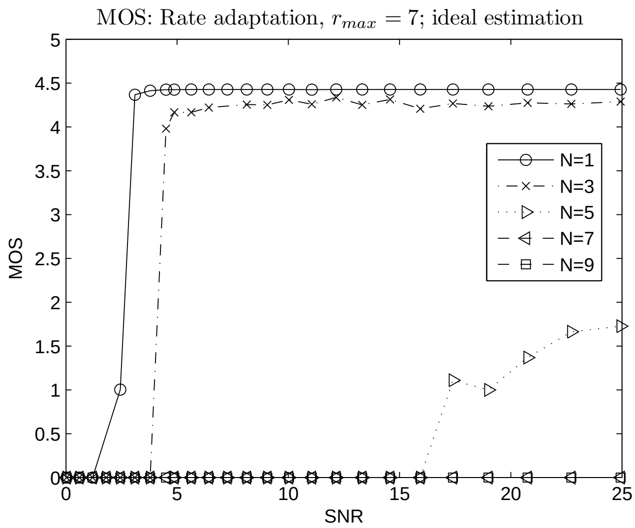

Figure 4.10: Only rate adaptation is performed in this case. An ideal estimation of the medium status is assumed. The maximum retry limit is set to $r_{max} = 7$ as suggested by the standard. The achieved MOS is shown as a function of the SNR, with N interfering nodes.

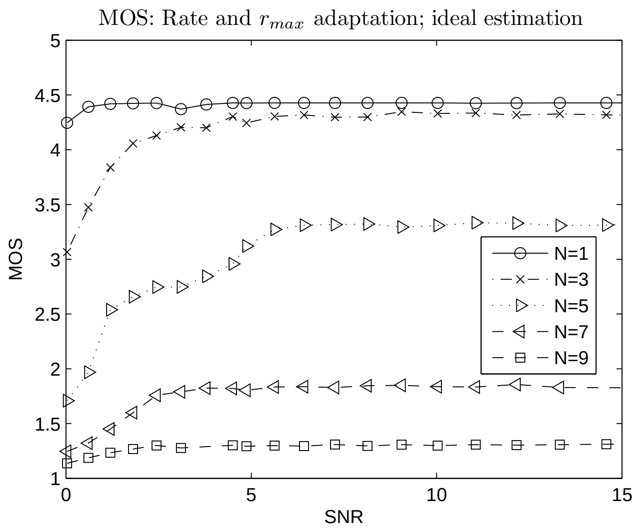

Figure 4.11: Both rate adaptation and maximum retry limit setting are performed in this case. An ideal estimation of the medium status is assumed. The maximum retry limit is automatically set by the algorithm which can choose a value $1 \leq r_{max} \leq 10$. The achieved MOS is shown as a function of the SNR, with N interfering nodes.

Figure 4.12 shows the same scenario where both rate adaptation and retry limit adaptation is performed, but in this case the medium status is estimated by using the MIB counters and following the procedure described in Section 4.3.

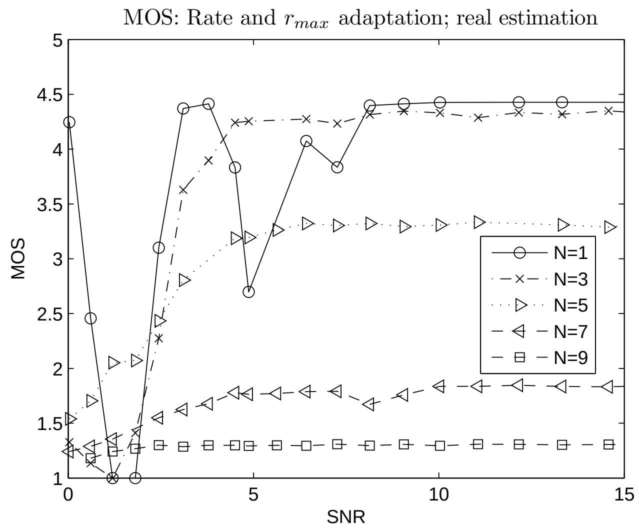

Figure 4.12: Both rate adaptation and maximum retry limit setting are performed in this case. The medium status estimation is performed based on the information provided by the MIB counters. The maximum retry limit is automatically set by the algorithm which can choose a value $1 \leq r_{max} \leq 10$. The achieved MOS is shown as a function of the SNR, with N interfering nodes.

The optimization still reaches a significant improvement over the simple rate adaptation. Some bad performance in the case of a single interferer can be explained by recalling that the status estimation and the successive parameter optimization assumes that the behavior of the mobile station which is performing the optimization do not impact the medium status and thus the behavior of the other stations. This is clearly not true in the case where a single interferer is present, and the proposed mechanism turns out to be sensible to this approximation in some scenarios. The solution of this problem is open for further research.

The algorithm has not been compared with preexisting schemes, since to the best of our knowledge no similar algorithms with the same objective exists and thus a fair comparison is not possible.

4.7 Conclusions

In this chapter a cross layer framework for optimizing the performance of an IEEE 802.11 WLAN connection has been presented. A Medium Status Estimation has been designed which allows to estimate propagation and interference conditions where the link under optimization is going to operate. Such quantities are estimated by using locally available information explicitly described in the standard specifications. A mathematical model for the link performance is then presented which allows to compute delay and packet loss probability as a function of the medium status and parameters setting. This model is used to define two optimization problems.

A Goodput Optimal Rate Adaptation algorithm, GORA, is designed which is able to adapt the transmission rate to the channel conditions and to the congestion caused by interfering nodes. In particular the ability of adapting the rate to the congestion conditions of the network, determined by the collision probability, is one of the key points in the proposed algorithm, which can be used as a reference benchmark in a wide range of operating conditions.

The framework is also used, by means of defining a different objective function, to optimize the network settings in order to maximize the quality of a VoIP connection. The Mean Option Score is chosen as the performance metric and has been mathematically related to some MAC parameters in order to define an optimization problem. In particular both the case of rate adaptation and the case of joint rate adaptation and retry limit adaptation are tested. Results shows how adapting the retry limit r_{max} is of fundamental importance for improving the performance of a VoIP connection.

4.8 Acknowledgments

The work presented in this chapter is a joint work with Federico Maguolo, Nicola Baldo, Andrea Zanella (SIGNET group at University of Padua) and in collaboration with Diego Melpignano and David Siorpaes (ST Microelectronics, Agrate Brianza). Thanks to Matteo Trivellato for the packet loss function in Figure 4.3.

The work presented in this chapter can be found in [C8] and [C9]. Patents are pending on the proposed algorithms.

Chapter 5

Resource management in FDMA cellular networks

Cellular networks based on OFDMA access appear as very promising systems to provide end-users with broadband wireless access. However, they also pose interesting challenges in terms of radio resource management. For this reason, we focus on scheduling at the LL Layer and resource allocation at MAC-PHY layers, and we investigate how they could be operated in a cross-layer fashion. A mechanism for a simple interaction between the layers is proposed in order to address both QoS requirements at the LL layer and efficiency at the PHY layer. The tunable mechanism, which is tested accounting for the interference in a multicellular scenario, allows to show the trade off between fairness requirements at the LL later and efficiency at the PHY layer. Due to the similarity of our system with the one described by the IEEE 802.16 standard, the results here depicted are also useful in the context of the standard system.

The work here presented has been developed within the FIRB/PRIMO project.

5.1 Introduction

In this chapter we discuss the challenge represented by packet scheduling and resource management through the realization of a joint scheduler/resource allocator. The reference scenario is represented by a multicellular system based on FDMA-TDD. In particular the downlink phase is considered and thus the described mechanisms will be applied at the base station side. We do not investigate the optimization issue, focusing instead on the possible choices for effective and simple implementation. The first contribution is the outline of a modular scheme where a credit-based scheduler is integrated with an efficient resource allocator based on a low-complexity power-efficient and capacity-driven heuristic criterion. Although these algorithm components are not themselves new, the original contribution is their integration in a modular framework, which represents one of the outcomes of the research project PRIMO [110]. As will be shown in the following, the proposed scheme

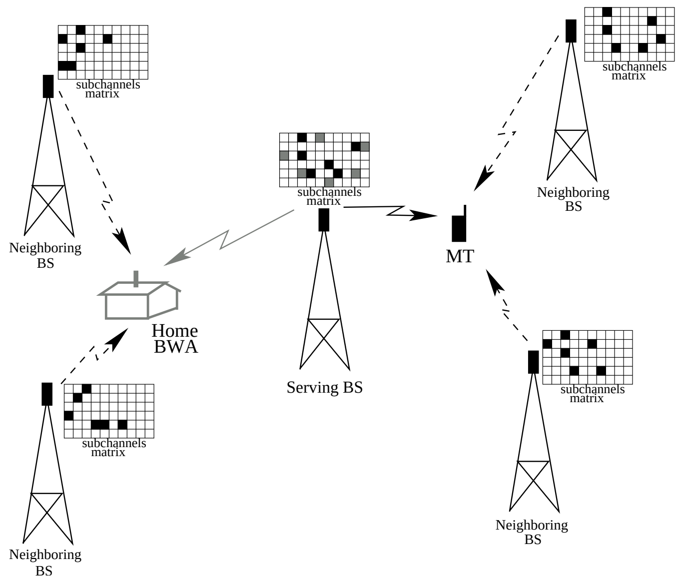

Figure 5.1: System scenario

is also able to provide tunability for what concerns the trade-off between overall power expenditure and time to achieve fairness among the users. This is possible by regulating the degree of freedom in the allocation, which is managed through a proper information exchange between the two modules of scheduling and resource allocation.

Another contribution of this research is the analysis of interference issues arising in such a scenario. In fact, even though there are studies proposing similar approaches, where packet scheduling and resource management are jointly addressed to have efficient solutions, for most of them the analysis and the performance evaluation are conducted in a simplified single-cell scenario. In these studies, the impact of other interfering cells is neglected, whereas in the system under examination full frequency reuse is envisioned and therefore the allocation of packets may be greatly affected by the interference conditions in the assigned resource. In this context, the outcome of simulations obtained with realistic models for the details of OFDM and the physical propagation scenario are presented and discussed, explicitly considering multi-cell interference.

For ease of presentation, in the following we first describe the considered system, focusing on the functional blocks and neglecting the implementation details, which will be provided in the simulation results section.

Then, the framework considered for the algorithm development is introduced together with the proposed scheduler, the resource allocator and the mechanism for the interaction among them. Finally, the simulation environment is described with the system implementation details and numerical results are shown, supporting our general conclusions on the relevance of channel state and interference awareness.

5.2 System model

5.2.1 MAC layer

The resource access in each cell is organized as a hybrid OFDMA-TDMA, using TDD duplex.

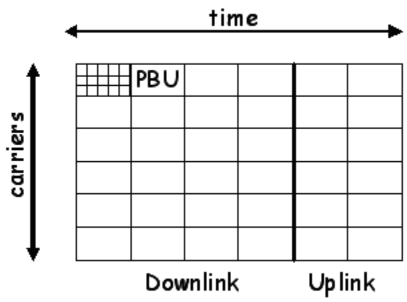

Figure 5.2: MAC structure

OFDM symbols are grouped in frames of duration T_f. Within each frame, symbols are devoted to control signaling, downlink or uplink transmissions. Figure 5.2 depicts the MAC frame structure that will be used in the system model.

Depending on the system configuration, the switching point between downlink and uplink transmissions within the MAC frame may dynamically change from frame to frame, according to different downlink/uplink traffic needs. We do not address this issue, assuming all the useful carriers are used for downlink transmissions.

Bandwidth is divided into N_s subcarriers and a frame contains N_t subsequent useful OFDM symbols. In order to reduce the resource addressing space, channel coherence in frequency and time is exploited by grouping adjacent $\langle$subcarrier, time–slot$\rangle$ pairs to form a logical subchannel, which is the minimum allocable unit of resource, denoted as Basic Transport Unit (BTU) in the following. The addressing space is thus reduced to M_s subchannels in frequency and M_t slots in time. Details on the MAC and OFDM parameters will be provided in Section 5.4.1.

Each BTU can be assigned to a different user and can be independently bit and power loaded. A fixed and discrete set of allowed bitload $B \in \{B_{min}, \ldots, B_{max}\}$ is defined for loading each BTU. In order to make use of this flexibility in the resource management, channel and interference measurements need to be exploited by scheduling and allocation algorithms.

We assume that in each frame the BS has perfect knowledge of the channel status and interference value for each subchannel of each user, as measured in the previous frame. This can be obtained, e.g., by piggybacking such information in each uplink packet. Due to the dynamics of the propagation environment and of the interference scenario, such information represents only an estimate of the channel status in the upcoming frame. Thus, the proposed framework is suitable for slowly varying channels.

5.2.2 LL Layer

Traffic produced by users is queued at the LL layer. Each user holds a separate queue which is loaded by a CBR traffic source. We assume the users' packets are further fragmented so as to produce smaller Data Chunks (DC) that can be entirely fitted in a single BTU. Thus we define the DC as a data packet of the same size of the minimum (non empty) bit load allocable on a BTU. Based on this definition, and according to the available set of bit loads, a single BTU can carry multiple DC. This assumption will lead to an easier formulation of the scheduling and resource allocation algorithms.

At the beginning of each frame, the scheduling mechanism, selects the candidates DCs for the transmission, according to the scheduling policy which will be described later on. The selected DCs are candidate for the allocation on physical resources and thus the transmission to the corresponding mobile station. DCs allocated and sent to the destination are deleted from the LL queues, whereas DCs neglected from the allocator are kept in the LL queues and reconsidered for future scheduling.

5.2.3 Relations with real systems

Among the emerging technologies to provide broadband wireless access (BWA), IEEE 802.16 represents one of the most promising and attractive systems. Expected areas of application of the IEEE 802.16 technology include high-speed Internet access, public services, private networks and broadband backbone for regions where wireless coverage is limited and deployment of cables would be too expensive or impractical.

However, it is often believed that a very important scenario, and probably the first to appear exploiting IEEE 802.16, will be the provision of BWA for moderate mobility scenarios such as residential Internet connections or offices. This will mean to realize, by means of IEEE 802.16, a multi-cellular setting, where fixed access points play the role of Base Stations (BS). The IEEE 802.16 air interface standard [111] describes in detail the physical (PHY) layer, based on Orthogonal Frequency Division Modulation (OFDM). In particular, the OFDMA with Adaptive Modulation and Coding (AMC) mode of IEEE 802.16 which appears to be very close to the model used in this chapter.

This method uses adjacent subcarriers to realize subchannels, resulting in a hybrid FDMA/TDMA medium access scheme. When used with fast feedback channels it can assign a modulation and coding combination per subchannel, enabling "water-pouring" types of algorithms, and it can also be used effectively with an Adaptive Antenna System (AAS). Several issues about scheduling and resource allocation algorithms are intentionally left open to developers, which stimulates researchers to seek strategies capable of providing good performance in this sense. The work presented in this chapter can thus be considered suitable for being extended to actual IEEE 802.16 networks.

5.3 Proposed mechanisms

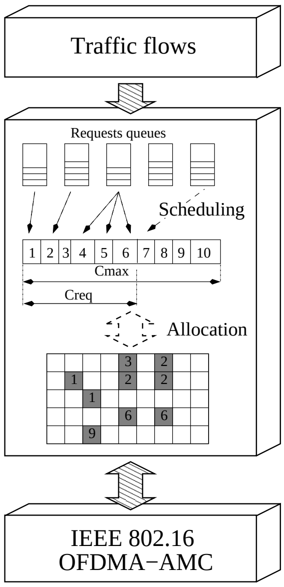

Figure 5.3: Proposed framework

In a cross-layer perspective, resource management could be pursued by a joint optimization of all the free allocation variables, i.e., which users to schedule, jointly with their power levels, subcarriers and timeslots, under QoS/fairness constraints for traffic scheduling and SNR/BER/power constraints at the physical layer. However, this would lead to a complex algorithm design, merging physical layer optimization goals with traffic level requirements and involving a large number of variables and parameters. This approach would also lose any flexibility if new traffic requirements or different optimization goals for the physical layer were to be considered, e.g., as a consequence of the introduction of new technologies.

Hence, our approach can be regarded to as a *loose cross-layer* one, trying to strike a balance between flexibility and modularity, as achieved by strict hierarchical layered design, and optimized performance, as could be yielded by cross-layer algorithms.

Radio resource allocation is aware of all physical layer constraints and can be given a desired physical transmission related optimization target. It aims at carrying a given traffic backlog of data units. Which DCs are the "hottest" or most valuable to carry and from which flow queue they should be taken is irrelevant to a radio resource allocator. This is the point where a traffic scheduler comes in. Our basic idea is to define the overall cross-layer resource management so that scheduling and allocation algorithms can be changed without impacting each other, provided that the common data structures are kept.

The coupling between scheduler and allocator is realized through a list of DCs, created according to the scheduling criteria, along with global parameters specifying handling constraints for those DCs. The list is defined frame by frame, and is processed by the allocator to define which input traffic to assign to that frame. After the allocation and the transmission, it is up to the allocator to update the status of the DCs of the current list as delivered, transmitted but failed, or not allocated at all. This feedback is used by the scheduler to update its own internal state and to provide a new list; the scheduler should also be given information about the maximum expected achievable capacity for each user, so as to make sensible scheduling decisions. Other common parameters, as detailed in the next two sections, make this a cross-layer approach, yet there is sufficient decoupling between DC scheduling and radio resource allocation algorithms so that they can be internally modified independently of each other. As an example, if the leading criterion for scheduling is changed from, say, non weighted fairness to delay deadline matching, this will affect the way the DC list is formed in each frame, but the allocation algorithm can be kept the same, as long as its objectives (i.e., minimize transmission power or achieved BER) make sense for the application scenario. The resource management architecture just described is schematically represented in Figure 5.3.

The scheduler and the allocator are two separate modules, whose implementation details are transparent as long as the interface between them remains the same. This differentiates the scheme here presented from other recent proposals, such as [112], where a joint MAC/PHY optimization is performed.

5.3.1 Scheduling algorithm

The scheduling algorithm itself (Algorithm 2) is a modified version of CBFQ [90]. For each flow (and thus for each user) we define a fixed weight ϕ_i and a credit counter K_i that increases when the flow is backlogged but not scheduled to transmit. At the beginning of each frame a list of DCs is tentatively scheduled; thus during the scheduling phase we consider temporary credit values K_i^*. The credit values K_i will only be updated after the allocator has selected the DCs which are actually going to be transmitted in the frame. The cycle in lines 5-21 is executed once per frame, and its purpose is to generate a list of at most C_{max} DCs to be sent to the allocator. The scheduler associates a priority value to each flow, based on its weight and on the credits it has accumulated. The highest priority flow is selected for scheduling and its temporary credit is decremented while the credit of all other flows is incremented. After the allocator has selected from the list the C_{req} DCs to be transmitted, the scheduler must update the credits by executing, for each chosen DC, lines 8-19 of the Algorithm 2 with K_i^* replaced by K_i.

Our goal is to fairly allocate the transmission resources to the flows according to their

Algorithm 2 schedule(C_{max})

for $i = 1$ to N **do**
  $K_i^* \leftarrow K_i$
  $c(i) \leftarrow 0$
end for
while $\sum_{i=1}^{N} c(i) < C_{max}$ **do**
  $f \leftarrow \arg\min_{i \in B} \frac{1-K_i^*}{\phi_i}$
  $c(f) \leftarrow c(f) + 1$
  for $i = 1$ to N **do**
    if $i \in B$ and $i \neq f$ **then**
      $K_i^* \leftarrow K_i^* + \max\left(\frac{1-K_f^*}{\phi_f}, 0\right) \cdot \phi_i$
    else if $i \notin B$ **then**
      $K_i \leftarrow 0$
    end if
  end for
  if $f \in B$ **then**
    $K_f^* \leftarrow \max\left(0, K_f^* - 1\right)$
  else
    $K_f^* \leftarrow 0$
  end if
  $SchedList$.**insert**(p_f)
end while
return $SchedList$

weight. An important property of our scheduling algorithm is that this fairness goal is attained independently of the algorithm used by the allocator. In the following we state a fairness bound [113, 114] by making no assumption as to the policy according to which C_{req} out of C_{max} DCs are selected for transmission. Let t be a frame index so that time is quantized in frames.

Theorem 1 *For two flows i and j continuously backlogged over an interval $[t_1, t_2)$ the following relation holds:*

$$\left|\frac{R_i\left(t_1, t_2\right)}{\phi_i} - \frac{R_j\left(t_1, t_2\right)}{\phi_j}\right| \leq \frac{C_{max} - C_{req} + 1}{t_2 - t_1}\left(\frac{1}{\phi_i} + \frac{1}{\phi_j}\right) \tag{5.1}$$

where $R_i\left(t_1, t_2\right)$ is the mean transmission rate (in packets/s) achieved by flow i over the time interval (t_1, t_2). Thus the (weighted) discrepancy between the transmission rates of any two flows can be made arbitrarily small by choosing a sufficiently long time interval.

It is then possible to show that Jain's fairness index [115], computed for N flows continuously backlogged over a time interval of m frames, is bounded as follows:

$$F(m) \geq \frac{1}{1 + 2\frac{N^2}{m^2}\left(\frac{C_{max}+1}{C_{req}} - 1\right)^2}, \; m \geq 2\left(\frac{C_{max}+1}{C_{req}} - 1\right) \tag{5.2}$$

5.3.2 Allocation algorithm

The allocation process consists in

- selecting C_{req} DCs out of C_{max}
- allocating the DCs on BTUs selecting the bit load and the transmission power

In the proposed framework, each cell performs its own resource allocation without explicit control information exchange with neighboring cells. The only used information refers to the channel and interference value measurements provided to the BS by its MTs.

As described in Section 5.3.1, the scheduling algorithm determines the aggregated throughput to be loaded on the cell and passes to the RRA a tentative list of data requests to be scheduled. Since the length C_{max} of this list (which determines the total available requests) can be greater than C_{req} (which determines the amount of requests that need to be allocated), RRA has a degree of freedom in selecting the subset of requests to be transmitted. Clearly, RRA selects such request subset in order to exploit the multiuser diversity.

Here we briefly describe the used allocation heuristic to allocate the NEW flows, with the aim of giving some insight on the problem and draw some interesting conclusions on the joint scheduling/allocation algorithms.

Let $c_{k,s,t}$ denote the Shannon capacity associated to BTU (s,t) when this BTU is used for a transmission toward the k-th MS with power $p_{k,s,t}$, $1 \leq s \leq M_s$ and $1 \leq t \leq M_t$). The capacity $c_{k,s,t}$ is

$$c_{k,s,t} = H_{BTU} \log_2 \left(1 + \frac{p_{k,s,t} G_{k,s,t}}{p_n + \sum_{i \in I} p_{i,s,t} \cdot G_{i,k,s,t}} \right) \tag{5.3}$$

where H_{BTU} is a scalar factor depending on the bandwidth and the time duration of a BTU, $G_{k,s,t}$ is the channel gain of MS k on the (s,t) BTU, p_n is the noise power and I is the set containing the BSs interfering over the (s,t) BTU. Let $b_{k,s,t} \in B$ be the bitload on the BTU (s,t) for request k. Since $c_{k,s,t}$ is the theoretical limit on the number of information bits that can be transmitted on the BTU (s,t), it is:

$$b_{k,s,t} \leq \alpha c_{k,s,t}, \tag{5.4}$$

where $\alpha < 1$ to take implementation limits into account. Let $\chi_{k,s,t}$ be the indicator function for the allocation of BTU (s,t) to request k, i.e.

$$\chi_{k,s,t} = \begin{cases} 1 & \text{if the } (s,t) \text{ PBU is assigned to user } k \\ 0 & \text{otherwise.} \end{cases} \tag{5.5}$$

Some constraints to the optimization problem need be added:

$$r_{k,min} \leq \frac{1}{T_f} \sum_{s=1}^{M_s} \sum_{t=1}^{M_t} b_{k,s,t} \chi_{k,s,t} \leq r_{k,max} \ , \tag{5.6}$$

i.e., the bit rate allocated to each request has to be within the imposed limits. Moreover, it is necessary to introduce a global constraint on the sum of all rates to be delivered, say r_{tot}. Without it, any allocation algorithm aiming at power minimization will allocate the minimum number of bits for each user.

$$\frac{1}{T_f} \sum_{k=1}^{K} \sum_{s=1}^{M_s} \sum_{t=1}^{M_t} b_{k,s,t} \chi_{k,s,t} = r_{tot} = C_{req} B_{min}. \tag{5.7}$$

A further constraint on the allocation is that, inside each cell, each BTU can be allocated to a single user only:

$$\sum_{k=1}^{K} \chi_{k,s,t} \leq 1 \quad \forall (s,t) \ . \tag{5.8}$$

Thus the allocation problem can be formulated as the problem of jointly finding the optimal set of values of $\chi_{k,s,t}$ (channel allocation), $b_{k,s,t}$ (bit loading) and $p_{k,s,t}$ (power loading) that enforces the constraints in (5.6)-(5.8) and optimizes a proper objective function (usually transmitted power). The complexity of this problem is very large and thus we propose a practical algorithm which solves a reduced and linearized version of it.

Efficiency maximization

The described problem requires a joint request selection, BTU allocation, bit loading and power loading [76]. Here a heuristic solution is proposed, which approximates the optimum solution with a reasonable complexity. The problem is solved by means of a greedy approach where the request selection, power loading, allocation and bit loading are solved in a disjoint way, by using four different heuristics.

In order to break the non-linear dependency among optimization variables, the request selection, allocation and bit loading problems have been solved after fixing a fictitious maximum power level $p_{k,s,t} = p \,\forall\, k, s, t$, and thus the Shannon capacity $c_{k,s,t}$, to a fictitious starting value for each BTU. Under this equal power allocation, the capacity $c_{k,s,t}$ is a direct measurement of the channel quality. Then, a metric η is associated to each user stating its goodness in terms of available capacity,

$$\eta_k = \sum_{(s,t)\in\mathcal{F}} c_{k,s,t},$$

where $\mathcal{F}$ is the set of unallocated BTUs.

At each step of the procedure, the request with highest η is selected for a BTU assignment. In the selection procedure, the requests that have not reached the minimum required rate are served first. Once the request to be served has been chosen, an efficiency metric ϵ is computed for each BTU as

$$\epsilon_{k,s,t} = \frac{c_{k,s,t}}{\sum_{i\in\mathcal{K}} c_{i,s,t}},$$

where $\mathcal{K}$ is the set of all the requests minus the request k. This index allows us to compare the advantage of allocating the BTU (s, t) to the request k, rather than to any other request. The BTU with the highest ϵ is associated to the request selected in the previous step. The effective bitload associated to the BTU is set to the highest possible value, less than the associated capacity. Eventually, the effective power value for each BTU is computed based on the actual loaded bits. The actual allocated power is increased by a constant factor in order to help overcome erroneous channel and interference estimation.

The procedure is repeated until all requests are satisfied, the aggregated requested rate is reached, there are no more free BTUs or the total allocated power has reached the threshold. Due to the per-BTU power limit, the requested aggregated throughput may not be reached. In this case the procedure is restarted using a higher value of $p^{i+1} = p^i + \delta p$. This algorithm can be efficiently implemented and shows a $O\left(KM_sM_t(\log(K) + \log(M_sM_t))\right)$ computational complexity.

This simple and low–complexity heuristic acts in a greedy way by decoupling the assignment of each allocation variable. Even though this is a suboptimal algorithm, it is shown to be able to exploit the multiuser and frequency diversity, and thus it is suitable for our purposes.

5.4 Simulations results

The simulator has been completely developed in C++, and has been realized by means of the definition of a set of interfaces and primitives to be used to allow the interaction between the MAC and PHY layers, coherently with the agreed cross-layer architecture.

5.4.1 Topology and propagation model

We consider the forward link of a multicellular system, where a complete reuse of the available time-frequency resources among neighboring cells is assumed.

Our simulator creates a cellular network scenario, with 9 BS distributed over a toroidal surface (this surface does not have edges, so that border effects are not present). The BSs are placed in a regular geometry with a minimum distance between two BSs of 1000 meters. The MSs are uniformly distributed and are associated each with the BS from which the best channel is sensed at the simulation start.

The assumed propagation model is derived from COST-259 [1].

For all three scenarios the same attenuation law will be applied, given by $\alpha(d) = \alpha_0 d^n$, where $\alpha_0$is a factor that takes into account the cell type, d is the transmitter-receiver distance in meters, and n is a parameter that in this study will be fixed equal to 4.0. In order to take into account the shadowing effects, log-normal shadowing with a standard deviation of 6 dB will be included. Hence, the propagation model used will have a distance-dependent path loss with a decay factor of 4.0 and log-normally distributed random path loss with a 6 dB standard deviation. The effect of multipath fading will be modeled by a sum of weighted delta-functions:

$$h(t, t') = \sum_{i=1}^{N} \beta_i \delta \left(t - t' - \frac{iT}{Q} \right)$$

where i is the path number, δ is the Dirac impulse, β and $\frac{iT}{Q}$ are the time-variant gain and delay of the path, respectively. The actual number of paths N varies depending on the value of the RMS (root mean square) delay spread σ. β_i are zero mean independent complex Gaussian stationary processes with an exponential decaying profile described by $Fe^{\frac{-iT}{Q\sigma}}$, where Q is the oversampling factor, T is the sampling period equal to 50 ns ($T = \frac{1}{B}$, where B is the system bandwidth) F is a normalizing factor.

The processes β_i have classical Jakes spectrum with Doppler frequency f_D that depends on the speed of the user $f_D = \frac{\nu}{c} f_c$, where ν is expressed in km/h and it is assumed f_c=5 GHz . The considered value for ν is of 1 m/s.

Since robust coding techniques are supposed to be employed at the PHY layer, we evaluate the probability of packet error according to the measured values of signal and interference strength on all the BTUs used to carry a packet.

In particular we compare the number of bits that could have been transmitted according to the experienced Shannon capacity (which derives from the actual values of SIR) with the loaded bits. If the capacity turns out to be greater than the loaded bits, the packet is assumed to be transmitted correctly.

5.4.2 MAC specifications

We set the BTU structure so as to mimic one of the possible settings allowed in the IEEE 802.16 standard.

The frame duration has been fixed to 5 ms. According to the 802.16 standard this is compatible with a frame subchannel structure which consists of 16 AMC subchannels (BTUs in our notation) in the frequency domain and 24 in the time domain. In this case each subchannel consists of 24 adjacent data subcarriers and two adjacent symbols.

The use of only one AMC format has been considered, corresponding to 144 bits per BTU.

For simplicity, in order to test the algorithm's behavior, the BS has been assumed to possess perfect channel and interference information for its MSs. The information is sent at the beginning of each frame, and refers to the measurements performed during the previous frame. To implement this exchange, several solutions are possible. In fact, the IEEE 802.16 standard provides several ways for the MS to send control information to the BS, allowing many levels of quantization and timing. The optimality of the allocation is a function of the amount of channel and interference information. Some preliminary evaluations of our scheme have shown that the overall signaling overhead, including this information as well as the broadcast of the schedule at the beginning of each frame, is limited to only a few percent.

5.4.3 Numerical results

A first consideration on the behavior of the joint scheduler/RRA refers to its ability to exploit the diversity naturally present in the system. In order to test whether the proposed framework is capable of taking advantage of the system multiuser diversity, in Fig. 5.4 results for a scenario with a variable number of users are reported. A constant global amount of traffic requests, independent of the number of users, has been passed to the allocator. The behavior has been tested under two different values of the parameter C_{max}.

Fig. 5.4 shows that the average transmission power decreases as the number of users increases. The proposed algorithm selects the flows to be scheduled by taking into account their channel state, thus exploiting multi-user diversity. As a consequence, the greater the number of users, the greater the chance of being able to schedule a subset of users who are all experiencing good channel conditions, which results in lower transmission power requirements. If we only pursued efficiency, with no regard for fairness, the power

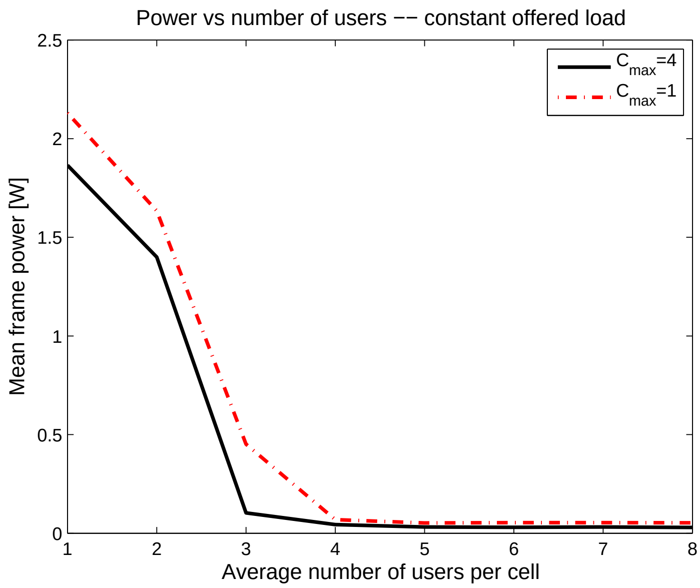

Figure 5.4: Average cell power consumption vs number of users, for two values of C_{max}

reduction with a large number of users would be even more significant: only the best users would be allowed to transmit, while the others would be permanently blocked. This behavior is confirmed by the fact that the power consumption with $C_{max} = 1$ is higher than in the case when $C_{max} = 4$. It is remarkable that our algorithm manages to achieve fairness among flows after a moderate number of frames, while succeeding in exploiting diversity with an intelligent time scheduling policy.

In the next set of results, we compare the power consumption and the fairness properties in order to point out that a trade off exists and that the behavior of the proposed algorithm can be tuned by acting on the parameter C_{max}.

We quantify the achieved throughput fairness among traffic flows in terms of Jain's fairness index [115], defined as $F = \left(\left(\sum_{i=1}^{N} x_i\right)^2\right) / \left(N \sum_{i=1}^{N} x_i^2\right)$, where x_i is the throughput achieved by flow i, and N is the number of competing flows. An index equal to 1 characterizes a perfectly fair outcome. We define the Time-To-Fairness metric (TTF) as the number of frames needed to reach a target fairness index, which has been fixed to 0.95.

Fig. 5.5 shows the mean transmission power and the TTF obtained with different values of C_{max} in a scenario with 6 users per cell on average, and a fixed required capacity. The traffic sources are assumed to be in saturation, i.e., each terminal always has packets to transmit. As expected the power decreases as C_{max} increases, while TTF increases. With $C_{max} = 1$ the allocator is forced to allocate all the requests selected by the scheduler, thus leading to strict fairness satisfaction but with a power–inefficient allocation; as C_{max} increases, the allocator has a higher degree of freedom and can choose to allocate only the best users, which results in a higher power efficiency, but also in a higher TTF.

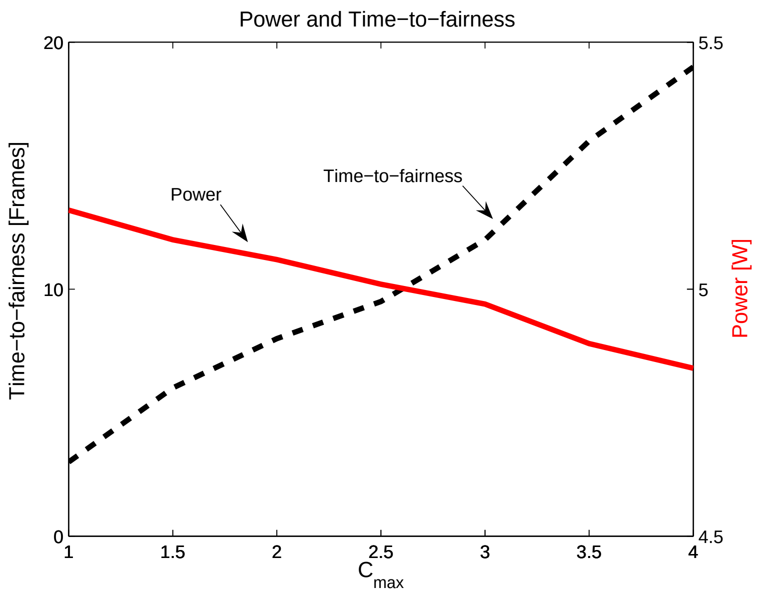

Figure 5.5: Average transmission power and TTF vs C_{max}, normalized to C_{req}.

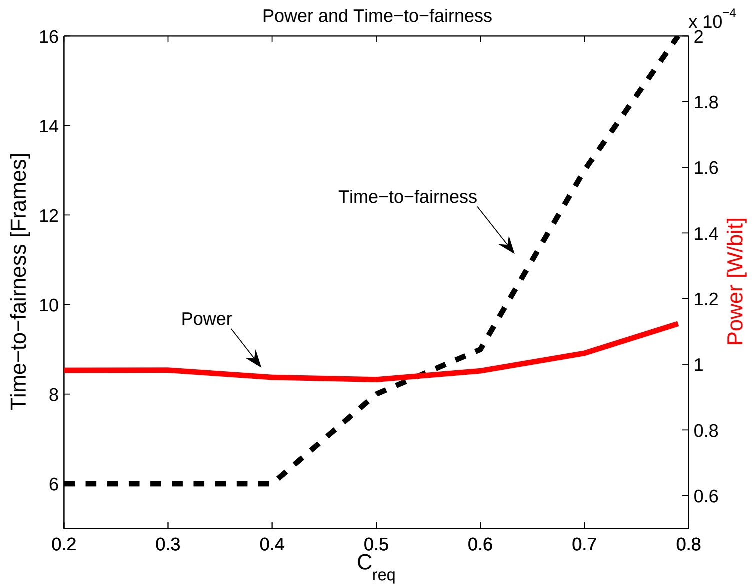

Figure 5.6: Average transmission power and TTF vs cell load

In Fig. 5.6 we plot the average power per transmitted bit and the TTF versus the normalized cell load C_{req}, with $C_{max} = 3$. The nearly constant per–bit power consumption observed with our allocator is an indication that an intelligent allocation policy, which also takes interference into account, is able to manage the inter cell interference by preferentially allocating less interfered resources. Anyway, as the cell load increases, the time needed to reach fairness increases.

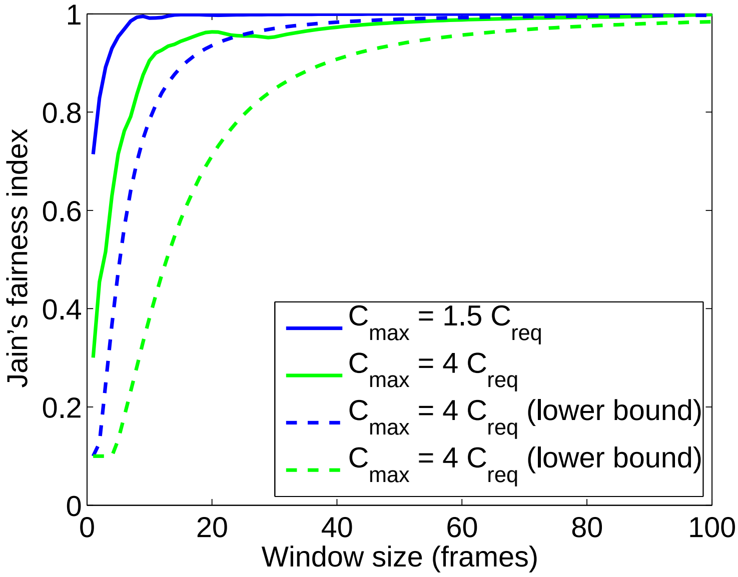

Figure 5.7: Fairness achieved after each frame

Fig. 5.7 shows Jain's fairness index, computed on a window of W frames, versus the window size W, thus describing the behavior of the algorithm through time. Two lower bounds are also shown. These lower bounds have been obtained by analytical formulas in [114], which are computed without taking into account the allocator policy, and are therefore valid for *any* resource allocation algorithm. The two curves, two different values of C_{max}, again show the effect of such parameter on the fairness outcomes. However, note also that the performance of the proposed strategy is very good if long-term fairness is considered.

5.5 Conclusions

In this chapter, we have discussed resource management aspects in a multicellular network based on OFDMA, where challenging optimization issues arise due to the complete resource reuse among neighboring cells. In this context OFDMA offers a flexible way to manage physical resources, leaving room for the development of efficient algorithms.

A framework for a dynamic resource management has been presented which merges traffic and physical level issues in a cross-layer perspective. Physical layer information, such as channel and interference measurements, together with traffic information, are exploited by the proposed two–layer algorithm in order to improve the performance.

In particular, simulations performed in a multicellular scenario have shown that the proposed joint scheduling and resource allocation algorithm is able to trade fairness requirements imposed at the flow level for physical efficiency metrics such as power consumption. The algorithm exploits the diversity naturally present in the system, thus confirming that FDMA resource management is an effective and promising approach for future generation wireless systems.

The studied system is similar to the one described in the IEEE 802.16 standard and thus can provide useful contributions for the study of that system.

5.6 Acknowledgments

The work presented in this chapter is a joint work with Silvano Pupolin, Zanella Andrea (University of Padua), Alfredo Todini, Andrea Baiocchi (INFOCOM Dept. Università la Sapienza, Roma), Leonardo Badia (IMT, Lucca, IT). The work has been developed within the FIRB/PRIMO project with the contribution of other researchers within the joint Work Packages 3 and 6. Related publications are [J4], [C3], [C4], [C5] and [C6].

Chapter 6

Further cross-layer investigations

As pointed out in the introduction, the concept of cross-layer optimization is quite general and can be applied to a broad variety of systems and used to approach many optimization issues.

During the Ph.D. course I had the chance to study very different issues in wireless networks involving the concept of cross-layer optimization. This chapter provides only a brief introduction to two additional topics, with a quick reference to the major results we found. Such work has been published in international journals and conferences, where the interested reader can find the details on each of them. A complete list of published/submitted papers is also provided.

6.1 Economic perspective in resource allocation

Related publications: [116] [117]

A very interesting and developed application of the Wireless Local Area Networks (WLANs) based on the IEEE 802.11 protocol is the creation of hot-spots, where a set of mobile terminals is connected to a central access point. This kind of system is nowadays present in business areas like conference rooms or airport and hotel lounges, where users are interested in easily and rapidly establishing a network connection.

Current implementations of IEEE 802.11 systems use the Distributed Coordination Function (DCF) based on the Carrier-Sense Multiple Access (CSMA). It is well known [105] that in this case the performance is heavily affected by the network operating conditions. Thus, the provider is interested in efficiently managing the bandwidth resource. Reasonably, this could mean aiming at achieving a satisfactory income from the network management operation and providing as many users as possible with a satisfactory service, which are required in order to have a sustainable economic model. For this reason, the investigations on how to properly allocate the radio resource, as well as to set up an appropriate pricing strategy, are key issues for the network operator.

To explore these aspects, we refer to the application of economic models to the Radio Resource Management, an open field for research on which several contributions have appeared in the recent literature. In particular, the concept of utility functions and issues taken from game-theory have been employed to represent a tunable Quality of Service (QoS), for example obtained through variations of the terminal's data rate.

An example of application of micro-economic issues to the management of a WLAN hot-spot is given in [118]. However, note that the micro-economic control performed there refers to the definition of a virtual price that has the effect of regulating the access and is negotiated dynamically [119]. Instead, in our study we considered the real price established by the operator for the service tariff, which is bound to be fixed *a priori* and known in advance by the users.

In particular, our aim is to investigate the role of actual pricing in determining resource usage. Besides causing revenue generation, pricing the system usage also allows a better coordination and a more efficient utilization. In other words, price tuning can be seen as an implicit Admission Control (AC) mechanism which improves the system performance. On the other hand, too high a price prevents users from entering the service, so that the system is under-utilized. Besides the total revenue, we also study the service appreciation by measuring the average number of satisfied users, which is another indicator of good management that a provider of a real system needs to take into consideration in the long run.

In order to perform these evaluations, we adopt the micro-economic model for wireless applications and services presented in [120], which describes the users' choices as driven by their appreciation of the service, and at the same time allows the evaluation of economic quantities such as the provider revenue and the average number of satisfied users.

Hence, our goal in this study is to apply such an economic model to a hotspot using CSMA/CA. As a closed formula for the performance of the joint microeconomic and network model is not available yet, we use NS2 simulations in order to test the proposed concept.

In this scenario, we aim at exploring the trade-off between flat and usage-based pricing, i.e., a purely linear function of the experienced data rate, or a hybrid one where price is only partially related to the achieved data rate. The comparison among different pricing strategies has to be based on some fairness criteria. We compare pricing strategies with equal average price (defined as the mean price experienced as a function of data rate).

The choice of a pricing strategy results in a different outcome of the network management, where, roughly speaking, users' satisfaction might be traded for provider's revenue.

As an example, in Figure 6.1 three scenarios are considered. Each curve represents the achieved $<$ operator revenue, user satisfaction $>$ as a function of a parameter which tune the pricing function from linear to flat. Each curve is obtained with a different average price.

As can be seen, usage-based policies (in this case, linear pricing) achieve higher revenue

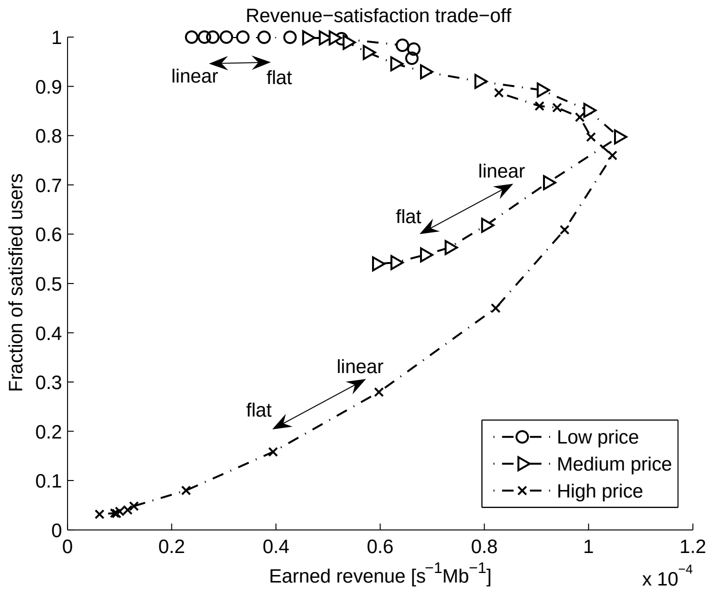

Figure 6.1: Example of results for the pricing policies evaluation through the micro-economic framework (the value of $\bar{p}$ is 0.3 for Low Price, 0.5 for Intermediate Price, 1.0 for High Price).

with respect to flat pricing but also yield lower users' satisfaction. This trade-off between the immediate goal of the provider and the users' welfare can be cut by an appropriate choice of the relative weight between the two contrasting objectives, so that in the end the choice of the pricing policy might be directly determined by looking at one suitable point in Fig. 6.1.

It is also worth noting that purely flat or purely linear strategies do not offer generally a good tradeoff, since the curves tend to wrap so that a hybrid strategy is often preferable. This emphasizes even more the need for an appropriate investigation of all pricing policies by allowing more factors than the simple average price in order to tune the price not only quantitatively but also qualitatively (i.e., changing the shape itself of the pricing function).

6.2 Efficient topologies in Bluetooth Scatternet

Related publications: [121] [122]

Originally born as a wireless replacement for cables connecting electronic devices, Bluetooth has been gaining a lot of consideration and attention by the scientific community in the last few years. The development of this technology is now focused on the area of the so-called Wireless Personal Area Networks (WPANs), where Bluetooth is expected to play a major role in the short and mid-term future. The commercial success of WPANs is intimately linked to their ability to support advanced digital services, like audio and video streaming, web browsing, etc. In such a scenario, the performance aspects of the radio technologies involved appear of primary importance.

Bluetooth has been designed to work in a scattered ad hoc environment, where multiple independent overlapping networks, called piconets, may coexist and be interconnected to form a multi-hop network, called *scatternet*. Recently, much attention has been devoted, by both academic and industrial world, to issues concerning scatternets formation and management. In particular, scatternet formation algorithms have been widely investigated, and many solutions have been proposed to build up a scatternet starting from disconnected units

The focus is now moving to the characterization and design of efficient scatternet topologies [123, 124, 125], since it is clear that the way piconets are interconnected to form scatternets may dramatically impact the network performance. The optimality of a scatternet configuration depends on the performance indexes considered. Some typical performance metrics are the number of piconets, the number of gateway devices, the number of roles per node etc. When data connections are considered, throughput, average and maximum traffic delay are taken as metrics of interest.

In particular we focus on the *network capacity*. This metric represents the supremum of the aggregated traffic that nodes can inject into the network without overflowing. In general, we say that a network configuration is *stable* if the total traffic offered to the network does not exceed the network capacity. In stable configurations, the average packet delay is almost surely finite and, provided that the dimensions of the buffers are adequately chosen, packets are never dropped because of overflows. Conversely, in unstable configurations some traffic connections will either experiment always increasing average packet delays or packet losses due to buffer overflows.

We investigate the relationship linking the scatternet configuration, i.e., the way piconets are interconnected to form the scatternet, and the network capacity. We propose a mathematical formalization of the notion of network capacity and show that it can be achieved in the presence of one-hop traffic patterns only. Thus, we discuss some topological conditions that are required to approach the network capacity, in the presence of local traffic only. This entail a formal justification of the inefficiency of tree topologies, and lead to the characterization of *closed-loop* as *capacity-efficient* (i.e., able to approach the capacity bound for any number of devices).

We also drop the assumption of traffic locality, in order to investigate the performance achieved by some specific scatternet topologies in the presence of a uniform traffic matrix, providing a sort of worst-case analysis. It is shown that some efficient solid configurations, based on Platonic solids, are able to outperform standard planar configurations in the presence of a limited number of devices. This analysis leads to the conclusion that *closed-loop* configurations possess some desirable properties, from both the point of view of network performance and ease of protocols implementation.

A general framework is also presented for the use of graph partitioning algorithms to arrange the nodes of a scatternet into a purposeful configuration. The application focus on closed-loop configurations, for which heuristics for the choice of master and gateway units is introduced and discussed. The generalization of the proposed techniques to account for spatial constraints (due to the fact that not all the nodes may be in mutual communication range) will also be discussed.

We address some optimization issues related to such configurations, showing how graph partition algorithms may be used to exploit traffic locality, enhancing the system performance. This work represents an attempt to provide mathematical insight into the relationship linking scatternet topology and performance. Our approach can be applied, for instance, to the design of networks of static sensors or domestic appliances, where the end-to-end traffic matrix among the nodes may be known *a priori*.

Chapter 7

Conclusions

In this thesis, the concept of cross-layer design in wireless networks has been discussed. The limits imposed by the layered networking reference model have been pointed out and illustrated with some examples. Thus, the idea of cross-layer design has been introduced as a useful paradigm for allowing the improvement of networks efficiency. To break the layered model is especially important in wireless networks, where the peculiar features of the transmission medium asks for complex optimization solutions.

Three main contributions have been presented in this thesis, which represent three examples of application of the cross-layer design concept to the wireless scenario.

The first one is referred to ad hoc networks where multiple channels are available and each node is provided with multiple interfaces. An analytical formulation of the optimization problem jointly addressing scheduling, channel allocation and congestion control has been presented. An iterative algorithm has then been developed which has analytically provable performance and gives useful insight for the design of practical solutions.

The second one addresses the problem of performance optimization in standard IEEE 802.11 networks. An optimization framework is introduced, which uses an estimation of the medium status and a mathematical model for the link performance with the aim of optimizing a given utility function. The framework allows for the definition of a Goodput Optimal Rate Adaptation Algorithm (GORA) and an algorithm for improving the performance of a VoIP connection. Both algorithms are based on analytical argumentations and are able to adapt the working parameters to both the propagation conditions and the interference in the network.

The third contribution refers to the proposal of a simple mechanism for a joint scheduling and resource allocation in the context of cellular networks. The proposed framework combines a fairness oriented scheduler with a heuristic resource allocation mechanism by means of an interface which allows for tuning the network working point by trading fairness for physical layer efficiency.

Finally, additional work in the context of performance optimization in wireless networks has also been briefly introduced, addressing microeconomic issues in resource allocation for WLANs and capacity related considerations in Bluetooth networks. The complete list of publications is provided in Appendix A.

Appendix A

Complete list of publications

Journals

[J5] L. Badia, S. Merlin, and M. Zorzi. A micro economic investigation of provisioning and pricing multimedia services over wireless LAN. *IEEE Transaction on Wireless Communications*, 2008.

[J4] L. Badia, A. Baiocchi, S. Merlin, S. Pupolin, A. Todini, A. Zanella, and M. Zorzi. On the impact of physical layer awareness on scheduling and resource allocation in broadband multi-cellular IEEE 802.16 systems. *IEEE Wireless Communications*, 1:36–43, 2007.

[J3] L. Badia, S. Merlin, A. Zanella, and M. Zorzi. Pricing VoWLAN services through a micro-economic framework. *IEEE Wireless Communications*, 13:6–13, 2006.

[J2] D. Miorandi, A. Zanella, S. Merlin, and A. Trainito. On efficient configurations for Bluetooth scatternet. *Ad-hoc networks, Elsevier*, 4:768–787, 2006.

[J1] A. Zanella, S. Merlin, and D. Miorandi. Mathematical analysis of the packet delay statistics in bluetooth piconets under round robin polling regime. *Mediterranean Journal on Computer and Networks*, 2:47–75, 2006.

Conferences

[C9] A. Zanella, N. Baldo, S. Merlin, F. Maguolo, M. Zorzi, D. Melpignano, and D. Siorpaes. APOS: Adaptive Parameters Optimization Scheme for voice over IEEE 802.11g. In *IEEE ICC*, 2008.

[C8] N. Baldo, F. Maguolo, S. Merlin, A. Zanella, M. Zorzi, D. Melpignano, and D. Siorpaes. GORA: Goodput Optimal Rate Adaptation for 802.11 using medium status estimation. In *IEEE ICC*, 2008.

[C7] S. Merlin, N. Vaidya, and M. Zorzi. Resource allocation in multi-radio multi-channel multi-hop wireless networks. In *IEEE INFOCOM*, 2008.

[C6] S. Merlin, L. Begnini, A. Zanella, L. Badia, and M. Zorzi. QoS-aware distributed resource allocation for hybrid FDMA/TDMA multicellular networks. In *WPMC*, 2006.

[C5] A. Todini, S. Merlin, A Zanella, A. Baiocchi, A. Valletta, D. Messina, and Tinnirello I. Allocation algorithms for PRIMO system. In *WiRTeP, Rome*, 2006.

[C4] A. Todini, A. Baiocchi, S. Merlin, A. Valletta, D. Messina, I. Tinnirello, B. Scanavino, and D. Veronesi. Cross-layer design of packet scheduling and resource allocation algorithms for 4G cellular systems. In *WPMC*, 2006.

[C3] S. Merlin A. Zanella. An efficient and adaptive resource allocation scheme for next generation cellular systems. In *WPMC*, 2005.

[C2] M. Borgo, A. Zanella, P. Bisaglia, and S. Merlin. Analysis of the hidden terminal effect in multi-rate IEEE 802.11b networks. In *WPMC*, 2004.

[C1] A. Zanella, G. Pierobon, and S. Merlin. On the limiting performance of broadcast algorithms over unidimensional ad-hoc networks. In *WPMC*, 2004.

Bibliography

[1] A. Eryilmaz, A. Ozdaglar, and E. Modiano. Polynomial Complexity Algorithms for Full Utilization of Multi-Hop Wireless Networks. *IEEE INFOCOM*, 2007.

[2] M. Kodialam and T. Nandagopal. Characterizing the capacity region in multi-radio multi-channel wireless networks. In *ACM MobiCom*, pages 73–87, 2005.

[3] Qi Wang and M. A. Abu. Cross-layer signalling for next-generation wireless systems. In *IEEE WCNC*, volume 2, pages 1084–1089, Mar. 2003.

[4] G. Carneiro, J. Ruela, and M. Ricardo. Cross layer design in 4G wireless terminals. *IEEE Wireless Communications*, 11(2):7–13, Apr. 2004.

[5] A. Ephremides and B. Hajek. Information theory and communication networks: an unconsummated union. *IEEE Transactions on Information Theory*, 44(6):2416–2434, 1998.

[6] D. Clark and D. L. Tennenhouse. Architectural considerations for a new generation of protocols. *ACM SIGCOMM Computer Communication Review*, 20(4), Sep. 1990.

[7] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson. Cross layer design for wireless networks. *IEEE Communication Magazine*, 41, Oct 2003.

[8] T. Goff, J. Moronski, DS Phatak, and V. Gupta. Freeze-TCP: a true end-to-end TCP enhancement mechanism for mobile environments. *IEEE INFOCOM*, 3, 2000.

[9] H. Balakrishnan, V.N. Padmanabhan, S. Seshan, and R.H. Katz. A comparison of mechanisms for improving TCP performance over wireless links. *IEEE/ACM Transactions on Networking*, 5(6):756–769, 1997.

[10] V. Kawadia and P.R. Kumar. A cautionary perspective on cross-layer design. *IEEE Wireless Communications*, 12(1):3–11, 2005.

[11] F.P. Kelly, A.K. Maulloo, and D.K.H. Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3):237–252, 1998.

[12] R. Srikant. A tutorial on cross-layer optimization in wireless networks. *IEEE Journal on Selected Areas in Communication*, 24(8):1452–1463, 2006.

[13] X. Lin and N.B. Shroff. Joint rate control and scheduling in multihop wireless networks. In *IEEE CDC*, 2004.

[14] L. Tassiulas, L. Georgiadis, and M. J. Neely. *Resource Allocation and Cross-Layer Control in Wireless Networks*. Now Publishers Inc., 2006.

[15] L. Chen, S. Low, M. Chiang, and J. Doyle. Cross layer congestion control, routing and scheduling design in ad-hoc wireless networks. In *IEEE INFOCOM*, pages 1–13, 2006.

[16] M.J. Neely, E. Modiano, and C.E. Rohrs. Dynamic power allocation and routing for time varying wireless networks. In *IEEE INFOCOM*, 2005.

[17] M. Chiang. Balancing transport and physical layer in wireless multihop networks: jointly optimal congestion control and power control. *IEEE Journal on Selected Areas in Communication*, 23:104–116, Jan 2005.

[18] R. Srikant. *The mathematics of internet congestion control*. Birkhauser, 2003.

[19] P. D. Palomar and M. Chiang. A tutorial on decomposition methods for network utility maximization. *IEEE Journal on Selected Areas in Communication*, 24(8):1439–1451, 2006.

[20] M. Chiang, S.H. Low, A.R. Calderbank, and J.C. Doyle. Layering as optimization decomposition: A mathematical theory of network architectures. *Proceedings of the IEEE*, 95(1):255–312, 2007.

[21] D.P. Palomar and M. Chiang. Alternative decompositions for distributed maximization of network utility: Framework and applications. In *IEEE INFOCOM*, 2006.

[22] S. Boyd and L. Vandenberge. *Convex Optimization*. Cambridge University press, 2004.

[23] X. Lin and N. B. Shroff. The impact of imperfect scheduling on cross-layer rate control in wireless networks. In *IEEE INFOCOM*, 2005.

[24] G. Zhang and Y. Liu. Stability and sensitivity for congestion control in wireless networks with time varying link capacities. In *ICNP*, 2005.

[25] Y. Yi and S. Shakkottay. Hop-by-hop congestion control over a wireless multihop network. In *IEEE INFOCOM*, 2004.

[26] P Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transaction on Information Theory*, 46(2), Mar. 2000.

[27] G. Sharma and N. Shroff. Maximum weighted matching with interference constraints. In *PERCOM*, 2006.

[28] S. Rasool and X. Lin. A distributed joint channel-assignment, scheduling and routing algorithm for multi-channel ad hoc wireless networks. In *IEEE INFOCOM*, 2007.

[29] E. Modiano, G. Zussman, and A. Brzezinski. Enabling distributed throughput maximization in wireless mesh networks via local pooling. Technical report, MIT, 2006.

[30] J.H. Hoepman. Simple Distributed Weighted Matchings. *Arxiv preprint cs.DC/0410047*, 2004.

[31] E. Modiano, D. Shah, and G. Zussman. Maximizing throughput in wireless networks with gossiping. In *ACM SIGMETRICS*, 2006.

[32] S. Basagni. Finding a maximal weighted independent set in wireless networks. *Telecommunication Systems, Springer*, 18(1):155–168, 2001.

[33] X. Wu, R. Srikant, and J. R. Perkins. Queue length stability of maximal greedy schedules in wireless networks. In *ITA Workshop*, 2006.

[34] P. Chaporkar, K. Kar, and S. Sarkar. Achieving queue length stability through maximal scheduling in wireless networks. In *ITA Workshop*, 2006.

[35] R. Srikant and X. Wu. Regulated maximal matching: a distributed scheduling algorithm for multi-hop wireless networks with node exclusive spectrum sharing. In *IEEE CDC*, 2005.

[36] N.H. Vaidya and P. Kyasanur. Multi-channel wireless networks: capacity and protocols. Technical report, Universtity of Illinois at Urbana Champaign, 2005.

[37] M. K. Marina and S. Das. A topology control approach for utilizing multiple channel in multi-radio wireless mesh networks. In *BroadNets*, 2005.

[38] A.P. Subramanian, H. Gupta, and S. R. Das. Minimum interference channel assignment in multi-radio wireless mesh networks. In *ACM MobiCom*, 2006.

[39] A.K. Das, HMK Alazemi, R. Vijayakumar, and S. Roy. Optimization models for fixed channel assignment in wireless mesh networks with multiple radios. *IEEE SECON*, pages 463–474, 2005.

[40] W. Wang and X. Liu. A framework for maximum capacity in multi channel multi radio wireless networks. In *ACM MobiHoc*, 2006.

[41] M. Shin, S. Lee, and Y.-A. Kim. Distributed channel assignment for multi-radio wireless networks. In *ACM MobiHoc*, 2006.

[42] A. Raniwala and T-C Chiueh. Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. In *IEEE INFOCOM*, 2003.

[43] M. Felegyahazi, M Cagalj, and J-P Hubaux. Multi-radio channel allocation in competitive wireless networks. In *IBC*, 2006.

[44] B-J Ko, V. Misra, J. Padhye, and D. Rubenstein. Distributed channel assignment in multi-radio 802.11 mesh networks. In *IEEE WCNC*, Hong Kong, March 2007.

[45] A. Mishra, S. Rozner, E. Banerjee, and W. Arbaugh. Exploiting partially overlapping channels in wireless networks: turning a peril into an advantage. In *IMC*, 2005.

[46] A. Mishra, S. Shrivastava, V. Banerjee, and W. Arbaugh. Partially overlapped channels not considered harmful. In *ACM Sigmetrics*, 2006.

[47] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *ACM MobiCom*, pages 114–128, 2004.

[48] P. Kyasanur and N. Vaidya. Routing and link layer protocols for multi-channel multi-interface ad-hoc wireless networks. Technical report, University of Illinois at Urbana Champaign, 2005.

[49] A Raniwala, K Gopalan, and T-C Chiueh. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8:50–65, April 2004. Invited paper.

[50] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *ACM MobiCom*, 2003.

[51] M. Kodialam and T. Nandagopal. Characterizing achievable rates in multi-hop wireless mesh networks with orthogonal channels. *IEEE/ACM Transactions on Networking*, 13(4):868–880, 2005.

[52] M. Alicherry, R. Bathia, and L. Li. Joint channel asignment and routing for throughput optimization in multi-radio wireless mesh networks. In *ACM MobiCom*, 2005.

[53] A. Mishra, V. Brik, S. Banerjee, and W. Arbaugh. A client driven approach for channel management in wireless LAN. In *IEEE INFOCOM*, 2006.

[54] P. Bahl, A. Adya, J. Padhye, and A. Wolman. Reconsidering wireless systems with multiple radios. *ACM SIGCOMM Computer Communications Review (CCR)*, 34(5), Oct. 2004.

[55] Z. Tang and J. Garcia Luna Acerves. Hop reservation multiple access (HRMA) for ad-hoc networks. In *IEEE INFOCOM*, 1999.

[56] A. Tzamaloukas and J. Garcia Luna Acerves. A receiver initiated collision avoidence protocol for multi channel networks. In *IEEE INFOCOM*, 2001.

[57] A. Nasipuri, J. Zhuang, and S. R. Das. A multichannel CSMA MAC protocol for multihop wireless networks. In *IEEE WCNC*, 1999.

[58] J. So and N. Vaidya. Multi channel mac for ad hoc networks: handling multi channel hidden terminals using a single transceiver. In *ACM MobiHoc*, 2004.

[59] J. Shin, P. Kishen, and T. La Porta. Dynamic multi-frequecy, multi-hop wireless cellular networks. In *MedHocNet*, 2006.

[60] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou. A multi radio unification protocol for IEEE 802.11 wireless networks. Technical report, Microsoft Research, 2003.

[61] K. N. Ramachandran, E. M. Belding, K. C. Almeroth, and M. Buddhikot. Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks. In *IEEE INFOCOM*, 2006.

[62] D. Qiao, S. Choi, and K.G. Shin. Goodput analysis and link adaptation for IEEE 802. 11 a wireless LANs. *IEEE Transactions on Mobile Computing*, 1(4):278–292, 2002.

[63] J. del Prado Pavon and S. Choi. Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement. In *IEEE ICC*, pages 1108–1113, 2003.

[64] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive MAC protocol for multi-Hop wireless networks. In *ACM MobiCom*, pages 236–251. ACM Press New York, NY, USA, 2001.

[65] A. Kamerman and L. Monteban. WaveLAN-II: A high-performance wireless LAN for the unlicensed band: Wireless. *Bell Labs technical journal*, 2(3):118–133, 1997.

[66] M. Lacage, M.H. Manshaei, and T. Turletti. IEEE 802.11 rate adaptation: a practical approach. In *ACM MobiCom*, pages 126–134. ACM Press New York, NY, USA, 2004.

[67] I. Haratcherev, K. Langendoen, R. Lagendijk, and H. Sips. Hybrid Rate Control for IEEE 802.11. In *ACM MobiWac*, pages 10–18, 2004.

[68] C. Hoffmann, MH Manshaei, and T. Turletti. CLARA: Closed-Loop Adaptive Rate Allocation for IEEE 802.11 Wireless LANs. *ACM Mobicom*, 1, 2005.

[69] J. Kim, S. Kim, S. Choi, and D. Qiao. CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs. *IEEE INFOCOM*, 2006.

[70] C. Boutremans and J-Y Le Boudec. Adaptive Joint Playout Buffer and FEC adjustement fot Internet Telephony. In *IEEE INFOCOM*, 2003.

[71] R.G. Cole and J.H. Rosenbluth. Voice over IP performance monitoring. *ACM SIGCOMM Computer Communication Review*, 31(2):9, 2001.

[72] A. Meddahi, H. Afifl, and D. Zeghlache. Packet-E-model: E-model for wireless VoIP quality evaluation. *IEEE PIMRC*, 3, 2003.

[73] J.S. Chow, J.C. Tu, and J.M. Cioffi. A discrete multitone transceiver system for HDSL applications. *IEEE Journal on Selected Areas in Commununications*, 9(6):895–908, Aug. 1991.

[74] T. Keller and L. Hanzo. Adaptive modulation techniques for duplex OFDM transmission. *IEEE Transaction on Vehicular Tech.*, 5:1893–1905, Sept. 2000.

[75] P. Viswanath, D. N. C. Tse, and R. Laroia. Opportunistic beamforming using dumb antennas. *IEEE Transaction on Information Theory*, Jun. 2002.

[76] C. Wong, R. Cheng, K. Letaief, and R. Murch. Multiuser OFDM with adaptive subcarrier, bit and power allocation. *IEEE Journal on Selected Areas on Communication*, 17:1747–1758, Oct. 1999.

[77] J. Cai, Shen X., and W. J. Mark. Downlink resource management for packet transmission in OFDM wireless communication systems. *IEEE Transaction on Wireless Communications*, 4(4):1688, 2005.

[78] I. Koutsopoulos and L. Tassiulas. Channel-state adaptive techniques for throughput enhancement in wireless broadband networks. In *IEEE PIMRC*, volume 2, pages 757–766, 2001.

[79] G. Song and Y. Li. Cross layer optimization for OFDM wireless networks. Part I: theoretical framework. *IEEE Transaction on Wireless Communications*, 4, 2005.

[80] G. Song and Y. Li. Cross layer optimization for OFDM wireless networks. Part II: algorithm development. *IEEE Transaction on Wireless Communications*, 4, 2005.

[81] G. Song and Y. (G.) Li. Utility-based resource allocation and scheduling in OFDM-based wireless broadband networks. *IEEE Communications Magazine*, 43(12):127–134, 2005.

[82] A. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The single-node case. *IEEE/ACM Transaction on Networking*, 1(3):334–357, 1993.

[83] H. Fattah and C. Leung. An overview of scheduling algorithms in wireless multimedia networks. *IEEE Wireless Communications*, 2002.

[84] T. Nandagopal, S. Lu, and V. Bharghavan. A Unified Architecture for the Design and Evaluation of Wireless Fair Queueing Algorithms. In *ACM MobiHoc*, pages 132–142, 1999.

[85] T. Nandagopal, S. Lu, and V. Bharghavan. A Unified Architecture for the Design and Evaluation of Wireless Fair Queueing Algorithms. *IEEE Wireless Networks*, 8:2331–247, 2002.

[86] T. S. Ng, I. Stoica, and H Zhang. Packet Fair Queueing Algorithms for Wireless Networks with Location-Dependent Errors. In *IEEE INFOCOM*, 1998.

[87] T. Nandagopal, S. Lu, and V. Bharghavan. A Wireless fair service algorithm for packet cellular networks. In *ACM MobiCom*, 1998.

[88] Azgin. A and M. Krunz. Scheduling in wireless networks under probabilistic channel information. In *ICCCN*, 2003.

[89] S. Lu, V. Bharghavan, and R. Srikant. Fair scheduling in wireless packet networks. *IEEE/ACM Transaction on Networking*, 1999.

[90] D. H. K. Bensaou, B. Tsang and T. Chan. Credit-based fair queueing (CBFQ): a simple service-scheduling algorithm for packet-switched networks. *IEEE/ACM Trans. on Networking*, 2001.

[91] Y. Liu, S. Gruhl, and E. W. Knightly. WCFQ: an opportunistic wireless scheduler with statistical fairness bounds. *IEEE Transactions on Wireless Communications*, 2003.

[92] X. Liu, E. K. P. Chong, and N. B. Shroff. A framework for opportunistic scheduling in wireless networks. *Computer Networks, Elsevier,* 41(4):451–474, 2003.

[93] M. A. HAleem and R. Chandranulti. Adaptive downlink scheduling and rate selection: a cross layer design. *IEEE Journal on selected Areas in Communication*, 6:1287, Jun. 2005.

[94] S. Borst. User-Level performance of channel-aware scheduling algorithms in wireless data networks. *IEEE Transaction on Network Computing*, 13(3):636, Jun. 2005.

[95] L. Tassiulas. Exploiting wireless channel state information for throughput maximization. *IEEE Transaction on Information Theory*, 50(11):2566, Nov. 2004.

[96] S. Shakkottay and L. Stoylar. Scheduling for multiple flows sharing a time varying channel: the exponential rule. Technical report, Bell Labs, 2000.

[97] Q. Liu, S. Zhou, and G. B. Giannakis. Cross layer scheduling with presribed QoS guarantees in adaptive wireless networks. *IEEE Journal on Selected Areas in Communication*, 23(5):1056, May 2005.

[98] Y. Liu and E. Knightly. Opportunistic fair scheduling over multiple wireless channels. In *IEEE INFOCOM*, 2003.

[99] P Kyasanur, J. So, C. Chereddi, and N. H. Vaidya. Multi-Channel Mesh Networks: Challenges and Protocols. *Wireless Communication*, 2005. invited paper.

[100] N. Jain and S. Das. A multichannel CSMA MAC protocol with receiver based channel selction for multihop wireless networks. In *IC3N*, 2000.

[101] A. Eryilmaz and R. Srikant. Joint Congestion Control, Routing, and MAC for Stability and Fairness in Wireless Networks. *Selected Areas in Communications, IEEE Journal on*, 24(8):1514–1524, 2006.

[102] ITU-T Recommendation G.107: The E-Model, a computational model for use in transmission planning, Dec 1998.

[103] ITU-T Recommendation G.113: General Characteristics of General Telephone Connectionsand Telephone Circuits - Transmission Impairments, Feb 1996.

[104] N. Baldo, F. Maguolo, M. Miozzo, M. Rossi, and M. Zorzi. NS2-MIRACLE: a modular framework for multi-technology and cross-layer support in Network Simulator 2 . In *ACM NSTools*, 2007.

[105] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, 2000.

[106] *ANSI/IEEE Std 802.11, 1999 edition*.

[107] G. Bianchi and I. Tinnirello. Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network. *IEEE INFOCOM*, 2, 2003.

[108] Andrea Zanella and Francesco De Pellegrini. Statistical characterization of the service time in saturated ieee 802.11 networks. *IEEE Communications Letters*, 3:225–227, 2005.

[109] A. Zanella, N. Baldo, S. Merlin, F. Maguolo, M. Zorzi, D. Melpignano, and D. Siorpaes. APOS: Adaptive Parameters Optimization Scheme for voice over IEEE 802.11g. In *Submitted to ICC 08*, 2008.

[110] Primo project firb, 2006. http://primo.ismb.it/firb/index.jsp.

[111] IEEE 802.16-2004. IEEE standard for local and metropolitan area networks – part 16: Air interface for fixed broadband wireless access systems, Oct. 2004.

[112] Y. J. Zhang and K.B̃. Letaief. Energy-efficient MAC-PHY resource management with guaranteed QoS in Wireless OFDM Networks. In *IEEE ICC*, 2005.

[113] A. Todini, A. Baiocchi, and A. Valletta. A scheduling algorithm for the OFDMA radio interface. Technical report, INFOCOM Department, Università La Sapienza, Roma, Jun. 2006. http://net.infocom.uniroma1.it/downloads/scheduler.pdf.

[114] A. Todini, S. Merlin, A Zanella, A. Baiocchi, A. Valletta, D. Messina, and Tinnirello I. Allocation algorithms for PRIMO system. In *WiRTeP, Rome*, 2006.

[115] R. K. Jain, D.M.W. Chiu, and W.R. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. *ACM Transactions on Computer Systems*, 1989.

[116] L. Badia, S. Merlin, and M. Zorzi. A Micro economic investigation of provisioning and pricing multimedia services over wireless LAN. *Transaction on Wireless Communications*, 2007.

[117] L. Badia, S. Merlin, A. Zanella, and M. Zorzi. Pricing VoWLAN services through a micro-economic framework. *IEEE Wireless Communications*, 13:6–13, 2006.

[118] R. Battiti, M. Conti, E. Gregori, and M. Sabel. Price-based congestion-control in Wi-Fi hot spots. In *WiOpt*, 2003.

[119] T. Heikkinen. On congestion pricing in a wireless network. *Wireless Networks*, 8(4):347–354, 2002.

[120] Leonardo Badia, Magnus Lindström, Jens Zander, and Michele Zorzi. Demand and pricing effects on the radio resource allocation of multimedia communication systems. In *Globecom 2003*, volume 7, pages 4116–4121, 2003.

[121] D. Miorandi, A. Zanella, S. Merlin, and A. Trainito. On efficient configurations for Bluetooth scatternet. *Ad-hoc networks, Elsevier*, 4:768–787, 2006.

[122] A. Zanella, S. Merlin, and D. Miorandi. Mathematical analysis of the packet delay statistics in bluetooth piconets under round robin polling regime. *Mediterranean Journal on Computer and Networks*, 2:47–75, 2006.

[123] R. M. Whitaker, L. Hodge, and I. Chlamtac. Bluetooth scatternet formation: a survey. *Ad Hoc Networks*, 3:403–450, Jul. 2005.

[124] M. Kalia, S. Garg, and R. Shorey. Scatternet structure and inter-piconet communication in the bluetooth system. In *IEEE National Conference on Communications, New Delhi*, 2000.

[125] K. E. Persson, D. Manivannan, and M. Singhal. Bluetooth scatternet: criteria, models and classification. *Ad Hoc Networks*, 2004.