UNIVERSIDAD CARLOS III DE MADRID

TESIS DOCTORAL

Analysis and Optimal Configuration of Distributed Opportunistic Scheduling Techniques in Wireless Networks

Autor: Andrés García Saavedra
Directores: Dr. Albert Banchs Roca
Dr. Pablo Serrano Yañez-Mingot

DEPARTAMENTO DE INGENIERÍA TELEMÁTICA

Leganés, Septiembre de 2013

TESIS DOCTORAL

Analysis and Optimal Configuration of Distributed Opportunistic Scheduling Techniques in Wireless Networks

Autor: Andrés García Saavedra
Director: Dr. Albert Banchs Roca
Director: Dr. Pablo Serrano Yañez-Mingot

Firma del tribunal calificador:

Firma:

Presidente:

Vocal:

Secretario:

Calificación:

Leganés, de de

"The signal is the truth.
The noise is what distracts us from the truth."
— *Nate Silver, The Signal and the Noise: Why So Many Predictions Fail - But Some Don't*

Acknowledgments

Me gustaría dar mi agradecimiento, especialmente, a Albert Banchs y Pablo Serrano. A Albert, primero, por haber confiado en mí y haberme ofrecido la posibilidad de trabajar en el grupo; y segundo, por su tenaz aportación de ideas, propuestas, consejos y conocimientos. A Pablo, por haber tenido siempre su puerta abierta para mí y por ser una fuente inagotable de motivación, ideas y consejos. En definitiva, a ambos por el equipo formado durante los últimos cuatro años, llenos de desafíos y excelentes resultados.

Quiero dedicar también unas líneas de agradecimiento a la Universidad Carlos III de Madrid y, en particular, al Departamento de Ingeniería Telemática y al grupo de investigación NETCOM por su financiación y apoyo. En especial a Arturo Azcorra por su confianza y la inestimable orientación brindada; a Alberto Gordillo, Carlos Jesús y Antonio, por su ayuda en cuestiones técnicas, docentes y de investigación; a Manolo, Rubén, Ángel, Marcelo, Ignacio, J. Alberto, Alberto, Jaime, María y resto de componentes del departamento que, de una u otra forma, han participado en mi experiencia docente y predoctoral.

I would like to particularly thank Gustavo de Veciana for his support during my stay in Austin. Even though it was a short experience, Gustavo's feedback, support and guidance during and after my visit are invaluable to me.

También me gustaría agradecer, de una manera muy especial, a Marco, J. Pablo, Paul y Chema por su amistad y su altruista disposición para ayudarme de una forma incansable; y a Isabel, Fabio, Andra y Roberto por haber formado parte de mi "día a día" y haberme ayudado en multitud de ocasiones.

A Laura por compartir su vida conmigo y ser también mi mejor amiga, por su paciencia y cariño y, sin duda, por ser el pilar más importante de mi vida.

Finalmente, a mis padres, a mis hermanas Ana y Tere, y al resto de mi familia. A Juan, Sergio, Efrén, Andrés J., Rubén, Raúl y Manuel. Porque han estado, están y estarán.

Esta tesis doctoral está especialmente dedicada a mi abuelo Amado, el molde que intento llenar.

Resumen

El fenómeno de "fading" o desvanecimiento en comunicaciones inalámbricas se ha considerado tradicionalmente como una fuente de problemas de fiabilidad que debe ser mitigada. En contraste, las técnicas de asignación de recursos oportunistas aprovechan las oscilaciones en la calidad de enlaces para mejorar el rendimiento global. Mientras que los mecanismos centralizados requieren una entidad central con información global, recientemente se han propuesto técnicas oportunistas distribuidas (DOS, por sus siglas en inglés) para operar en redes donde dicha entidad no está disponible, o donde el coste en la comunicación para proporcionarle información puntual es prohibitivo.

Con DOS, cada estación contiende por el canal con una cierta probabilidad. Si la contienda resulta exitosa, la estación mide la calidad del canal y transmite si ésta supera un cierto umbral. De lo contrario, la estación no aprovecha esa oportunidad para transmitir, permitiendo a todas las estaciones contender de nuevo. Dado que estaciones diferentes, en distintas instancias de tiempo, experimentan diferentes condiciones de canal, es probable que un enlace con mejores condiciones use el canal, mejorando el rendimiento global.

En esta tesis proponemos primero ADOS, un mecanismo adaptativo que lleva al sistema a un reparto óptimo de los recursos en términos de *equidad proporcional*. Mostramos que este mecanismo supera el rendimiento de trabajos previos, particularmente en escenarios con estaciones no saturados (que no siempre tienen datos que transmitir). La naturaleza distribuida de DOS lo hace particularmente vulnerable a usuarios *egoístas* que buscan maximizar su rendimiento a expensas de aquellos que cooperan por el *bien común*. Así, diseñamos un mecanismo, llamado DOC, que (i) optimiza el rendimiento si todos los nodos obedecen el protocolo, y (ii) elimina cualquier posible beneficio por desviarse del mismo (y así, el incentivo a no cooperar). Finalmente, proponemos un nuevo criterio de asignación de recursos, llamado EF, que supone un compromiso entre la configuración más eficiente energéticamente (donde todos los recursos se asignan a los nodos más eficientes) y una asignación donde todos comparten de forma equitativa los recursos sin tener en cuenta su consumo. Dada la falta de modelos para predecir de forma precisa el consumo de dispositivos inalámbricos, llevamos a cabo un estudio experimental que resulta en un modelo energético que completa a la literatura existente. Finalmente, aplicamos lo anterior para diseñar una estrategia que optimiza EF en redes basadas en DOS.

Abstract

The phenomenon of fading in wireless communications has traditionally been considered as a source of unreliability that needs to be mitigated. In contrast, Opportunistic Scheduling (OS) techniques exploit quick channel quality oscillations in fading links, during the assignment of transmission opportunities, to improve the performance of wireless networks. While centralized mechanisms rely on a central entity with global knowledge, Distributed Opportunistic Scheduling (DOS) techniques have recently been proposed to work in distributed networks, i.e., where either such a central entity is not available, or the communication overhead to feed timely information to this central entity is prohibitive.

With DOS, each station contends for the channel with a certain access probability. If a contention is successful, the station measures the channel conditions and transmits if the channel quality is above a certain threshold. Otherwise, the station does not use the transmission opportunity, allowing all stations to recontend. Given the fact that different stations, in different time instances, experience different channel conditions, it is likely that the channel is used by a link with better conditions, improving overall performance.

In this thesis we first propose ADOS, an adaptive mechanism that drives the system to an optimal allocation of resources in terms of *proportional fairness*. We show that this mechanism outperforms previous approaches, particularly in scenarios with non-saturated stations (that do not always have data to transmit). The distributed nature of DOS makes it particularly vulnerable to *selfish* users that seek to maximize their own performance at the expense of those that cooperate for the *common welfare*. We thus design a punishing mechanism, namely DOC, that (*i*) drives the system to the optimal point of operation when all stations follow the protocol, and (*ii*) removes any potential gain by deviating from it (and thus, the incentive to misbehave). Finally, we propose a novel allocation criterion, namely the EF criterion, to balance between the most energy-efficient configuration (where all resources are given to the most energy efficient devices) and the throughput-optimal allocation (where all devices evenly share the resources regardless of their power consumption). Due to the lack of models that accurately predict the power consumption behavior of wireless devices, we perform a thorough experimental study to devise a power consumption model that completes existing literature. Finally, we apply these findings to design an EF-optimal strategy in DOS networks.

Contents

List of Figures

Chapter 1

Introduction

Wireless communication is currently one of the most popular areas in the field of Information and Communication Technologies (ICT). The ever-growing increase in the demand for untethered online services, coupled with the explosive market success of mobile communication devices such as laptops, smart-phones or tablet computers, well explain this success. Moreover, projections of authority suggest a trend far from stabilizing [1], with a demand of mobile services that *doubles* every year, making the task of satisfying these users' needs very challenging.

Communication over wireless channels faces two fundamental problems inherent to the medium: *interference* and *fading*. First, unlike in wireline communications where we can create a dedicated link between transmitter and receiver, a wireless user utilizes the air as channel to transfer information, a medium subject to distorting interferences from other electromagnetic radiation sources (like other wireless users). Second, the presence of objects, that may be in movement, mobility in transmitters and/or receivers, and other irregularities in the surroundings cause three propagation phenomena [2]: reflection, diffraction and scattering. These lead to multiple paths for the propagation of several copies of the original signal to reach the intended destination, though with differences in amplitude, phase shift and delay. The superposition of these multiple copies makes the receiver experience fluctuations in the signal strength, that is, fading, at different timescales: *slow fading* for slow variations with respect to the transmission delay, and *fast fading* for rapid oscillations.

1.1 Opportunistic scheduling

Interferences have traditionally been tackled by Radio Resource Management (RRM) mechanisms, e.g., at the MAC layer through techniques such as CSMA/CA and RTS/CTS [3]. On the other hand, fading has largely been considered a physical layer problem

and is usually addressed with the objective of maximizing reliability, e.g., through proper selection of a coding and modulation scheme. However, the physical layer does not always hide fading effects from the MAC layer [4], and using very conservative channel coding and modulation schemes that may allow decoding during deep fades wastes capacity and energy. In contrast, in the last few years, the research community has shifted to a *more proactive* position; this novel point of view sees fading as an *opportunity* to exploit in order to increase the spectral efficiency.

In this context, opportunistic scheduling (e.g., [5, 6]) addresses the issue of channel quality variations by preferentially scheduling transmissions of senders with good instantaneous channel conditions. Exploiting knowledge of the channel conditions in this manner has been shown to lead to substantial performance gains. While centralized opportunistic scheduling mechanisms [5, 6] rely on a central entity with global knowledge of the radio conditions of all links, the more recent Distributed Opportunistic Scheduling (DOS) techniques [7–10], also work in settings where either such a central entity is not available, or the communication overhead to provide timely updates of the channel conditions of all the stations to the central entity is prohibitive.

With DOS, stations contend for channel access and, upon successful contention, a station gathers local information about the instantaneous channel conditions to decide whether to transmit data or give up the transmission opportunity. This decision is taken based on a pure threshold policy, i.e., a station gives up its transmission opportunity if the transmission rate allowed by the channel conditions falls below a certain threshold. By giving up a transmission opportunity and allowing recontention, it is likely that the channel is taken by a station with better channel conditions, resulting in a higher aggregate throughput. Furthermore, since no coordination between stations is required, DOS protocols are simpler to implement and have a lower control overhead compared to centralized opportunistic scheduling mechanisms.

The seminal work by Zheng et al. [7] lays the basic foundations of distributed opportunistic scheduling. The authors propose a mechanism based on optimal stopping theory [11] and analyze its performance with well-behaved as well as selfish users. The aim of the algorithm is to maximize the total throughput of the network. [8–10, 12] extend the basic mechanism of [7] by analyzing the case of imperfect channel information [9], improving channel estimation through two-level channel probing [8], and incorporating delay constraints [10]. In turn, [12] avoids the assumption used in [7] of independent observations during the probing phase proposing the idea of effective observation points. While the configuration strategies proposed in this thesis deal with the basic DOS mechanism of [7], it can be extended to incorporate the enhancements of [8–10, 12].

The contributions of [7–10] have provided valuable insights and a deep understanding of DOS techniques and their performance; however, less attention has been paid to

the design of the algorithms implementing DOS. In this thesis we address the issue of adaptively configuring DOS pursuing a set of different objectives, namely (*i*) provide a proportional fair allocation of rates among users, (*ii*) remove the incentive of *selfish* users to misbehave, (*iii*) maximize the energy efficiency of the system.

1.2 Fairness issues in heterogeneous scenarios

Scheduling schemes that opportunistically exploit the time-varying conditions of wireless links do improve the spectrum efficiency and achieve multiuser diversity gains. However, in the pursuit of maximizing the overall throughput, the performance of stations with poorer average channel conditions (e.g., because they are further away from the receiver) may be sacrificed. Many works have particularly considered this issue in order to provide Quality of Service (QoS) in broadband wireless networks (e.g,. [13–15]). However, the provision of fairness in distributed networks where users individually take their own scheduling decisions is indeed more challenging.

In order to solve the tradeoff between maximizing total throughput and a pure fair allocation, Kelly's *proportional fairness* criterion [16] is well accepted in the literature. This criterion was originally proposed in the context of wired networks, and has been widely used to address a variety of throughput fairness issues. This criterion is defined as follows. A throughput allocation $\{r_1, \ldots, r_n\}$ is proportionally fair if it is feasible, and for any other feasible allocation $\{r_1^*, \ldots, r_n^*\}$ the aggregate of proportional changes is not positive, i.e.,

$$\sum_i \frac{r_i^* - r_i}{r_i} \leq 0 \tag{1.1}$$

Note that, with the above definition, in a two station scenario the throughput of one station would be decreased by say 10% only as long as this allowed an increase in the throughput of the other station of more than 10%, which represents a balance between two extreme allocations (i.e., throughput is equally shared, or throughput is given to the most efficient station). To investigate the proportional fair allocation further, we consider a small perturbation around the proportional fair allocation $r_i \to r_i + dr_i$. From (1.1),

$$\sum_i \frac{dr_i}{r_i} \leq 0 \Longrightarrow \sum_i (\log(r_i))' dr_i \leq 0$$

It follows from the above that the proportional fair allocation represents a local maximum of the function $\sum \log(r_i)$. Since this is a concave function, it has only one maximum, and therefore the local maximum is also the global maximum. We can identify the proportional fair (PF) allocation with the one that maximizes the sum of the logarithms:

$$PF \Longleftrightarrow \max \sum \log(r_i)$$

In this thesis, we design an adaptive algorithm to drive the system to a PF-optimal point of operation. The advantages of our proposal is that it does not require explicit cooperation between nodes and works well under non-saturated conditions, i.e., where there are users that do not always have data ready to transmit.

1.3 The incentive to *misbehave*

The lack of global channel information makes DOS systems very vulnerable to *selfish* users: by deviating from the protocol and using more transmission opportunities, a selfish user can gain a greater share of wireless resources at the expense of *well-behaved* users. A number of works in the literature have addressed the selfishness problem in distributed wireless networks, particularly in the context of CSMA/CA protocols [17–23]. The lack of cooperation from these selfish users may drive the system, not only to an unfair share of resources, but also to network starvation; and makes the subject critical to be considered in the design of distributed algorithms.

In this thesis, we also address the problem of selfishness in DOS from a game theoretic standpoint. In our formulation of the problem, the players are wireless stations that implement DOS and strive to obtain as great a share of resources as possible from the wireless network. We show that, in the absence of penalties, the wireless network naturally tends to either great unfairness or network collapse. Building on this result, we design a penalty mechanism in which any player who misbehaves will be punished by other players in such a way that there is no incentive to misbehave. A key challenge when designing such a penalty scheme is to carefully adjust the punishment inflicted on a misbehaving station. If the punishment is too light, a selfish station may still benefit from misbehaving. If it is too excessive, however, the punishment itself could be interpreted as misbehavior and trigger punishment from other stations, leading to an endless spiral of increasing punishments and ultimately throughput collapse.

1.4 Energy efficiency

The increase in energy density of current state of the art (Lithium-Ion) batteries is far from following Moore's Law, the current challenge being "just" a twofold density increase in the next 10 years [24]. This is not a good technological premise behind the energy greediness of wireless connectivity, second only to that required to backlight displays in most handheld devices. Moreover, battery powered wireless devices are becoming ubiquitous, and are frequently part of the network infrastructure itself; even besides the obvious

case of wireless sensor networks, battery powered relays or opportunistic intermediaries are widely considered in ad hoc, mesh, DTN scenarios, or emergency deployments.

It is hence not nearly a surprise that a *huge* research effort has been dedicated to find ways for reducing energy consumption in the wireless access and communication operation [25,26]. For instance, with reference to the 802.11 WLAN (WiFi) technology [27] energy efficiency improvements span very diverse aspects of the 802.11 operation, from management procedures [28, 29], to usage of opportunistic relays [30, 31] or infrastructure on demand [32], to PHY [33] and MAC [34, 35] layer parameters' optimizations, and so on.

In this line, opportunistic scheduling techniques can also be applied to pursue an energy-efficient allocation of resources; thus, we study in this thesis how to exploit DOS techniques to maximize bits per joule instead of bits per unit of time. It is, however, not clear how to achieve this in a network mostly comprised of heterogeneous devices, indeed the most common scenario in real life, as the optimization of *just* overall energy efficiency might end up *choking* the least efficient nodes. Inspired by the Proportional Fairness (PF) criterion mentioned above, we propose the Energy-efficient proportionally Fair (EF) criterion to balance these two extreme configurations, i.e., a configuration that allocate resources to the most energy-efficient devices to maximize overall performance, and another configuration that does not consider energy consumption to schedule transmission opportunities.

A fundamental problem that we face while addressing the aforementioned criterion is the lack of accurate models to account for the per-frame energy consumption of wireless devices. To overcome this issue, we propose in this thesis an accurate model based on a thorough experimental study. This study reveals a "new" energy cost associated to the software processing of each frame that is not accounted for by related literature and it is definitely not negligible.

Finally, building on top of these findings we utilize optimal stopping theory [11] to exploit DOS in order to achieve an EF-optimal allocation of resources in wireless distributed networks.

1.5 Summary of thesis contributions

The contributions of this thesis are summarized as follows. First, we find the optimal configuration strategy for a wireless distributed network that exploits opportunistically the fading phenomenon (i.e., namely a DOS system) to achieve a proportional fair allocation of the throughputs, that is, a configuration that provides a good tradeoff allocation between total throughput and fairness. We then propose ADOS [36,37], an efficient adaptive mechanism, based on control theory, that drives the system to the optimal point of

operation above. A key advantage of the mechanism over previous proposals is that it performs well in wireless networks with non-saturated stations[1]. The analysis and design of previous approaches requires the assumption that all stations are always saturated, resulting in overly conservative behavior under non-saturation conditions. In contrast, our approach adapts to the *actual* network load instead of the number of stations, and hence naturally lends itself to non-saturated conditions. An additional advantage of the proposed mechanism is that it only relies on information that can be observed locally, in contrast to previous approaches which require global information and hence incur a substantial signaling load to gather this information. Finally, ADOS also adapts quickly to changes in the network such as nodes entering and leaving and/or moving stations.

Second, we perform a game theoretic analysis of DOS considering that users can selfishly configure both their access probability and transmission rate threshold, whereas previous analysis assume that selfish users only have control over the thresholds. We then propose DOC [38], a distributed penalty mechanism to force an optimal Nash equilibrium. DOC has been carefully calibrated to (*i*) lead the system to the optimal configuration proposed above and (*ii*) penalize, quickly but stably, those stations that choose a selfish configuration with the goal of perceiving more throughput.

Third, we look into the configuration of DOS to improve energy efficiency rather than throughput performance in wireless networks. We show how a *pure* energy-efficient strategy can cause great unfairness in the allocation of the rates; hence we propose a criterion strategy, namely Energy-efficient Fair criterion (EF), that balances throughput fairness and energy-efficiency in a wireless network [39, 40]. The application of energy-aware optimization criteria requires reliable models that capture the power consumption behavior of wireless devices. As we show in this thesis, related work on this topic lack of meaningful factors that render existing models inaccurate for our purposes. To address this problem, we perform a thorough experimental study of the power consumption of wireless devices and unveil a novel model [41, 42], that accurately predicts the per-frame power consumption of wireless transceivers. Building on top of these findings, we finally use optimal stopping theory to propose an energy-efficient strategy that provides an EF-based optimization in DOS [43].

1.6 Thesis overview

The remainder of this thesis is organized as follows. Chapter 2 presents the relevant literature related to this thesis. In Chapter 3 we first introduce the system model description of the Distributed Opportunistic Scheduling (DOS) network under study. Then, we

[1] A saturated station always has data ready for transmission while a non-saturated station may at times have nothing to send.

devise the optimal configuration strategy that provides a proportional-fair (PF) allocation of individual throughputs, and we design an adaptive algorithm to drive the system to such optimal point of operation. A network populated with *rational* users that selfishly want to maximize their individual utility naturally tends to starvation if the operative algorithms do not impose any penalization mechanism. In this context, we propose in Chapter 4 an adaptive algorithm that drives the system to the PF-optimal configuration while it also removes the incentive of selfish users to misbehave. We look for an energy efficient configuration strategy for DOS-based networks in Chapter 5. The lack of accurate models that predict the per-frame power consumption of wireless devices is addressed by our experimental study in this chapter. Then, given our power consumption model, we motivate the importance of considering throughput fairness when aiming at maximizing overall energy efficiency and propose a criterion to balance the two performance figures. Finally, we show how to apply this criterion in DOS using optimal stopping theory. Chapter 6 concludes this dissertation with the final remarks and future lines of work.

This thesis covers contributions from the following literature:

- A. Garcia-Saavedra, A. Banchs, P. Serrano, and J. Widmer, "Distributed Opportunistic Scheduling: A Control Theoretic Approach," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, Orlando, FL, March 2012 (Chapter 3).

- A. Garcia-Saavedra, A. Banchs, P. Serrano, and J. Widmer, "Distributed Opportunistic Scheduling: A Control Theoretic Approach," *Submitted* (Chapter 3).

- A. Banchs, A. Garcia-Saavedra, P. Serrano, and J. Widmer, "A Game-Theoretic Approach to Distributed Opportunistic Scheduling," *IEEE/ACM Trans. on Networking*, Accepted (Chapter 4).

- A. Garcia-Saavedra, P. Serrano, A. Banchs, and G. Bianchi, "Energy consumption anatomy of 802.11 devices and its implication on modeling and design," in *Proc. of ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, Dec. 2012, pp. 169–180 (Chapter 5).

- P. Serrano, A. Garcia-Saavedra *et al.*, "Per-frame Energy Consumption in 802.11 Devices and its Implication on Modeling and Design," *Submitted* (Chapter 5).

- A. Garcia-Saavedra, P. Serrano, A. Banchs, and M. Hollick, "Energy-efficient fair channel access for IEEE 802.11 WLANs," in *Proc. of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2011 (Chapter 5).

- A. Garcia-Saavedra, P. Serrano, A. Banchs, and M. Hollick, "Balancing energy efficiency and throughput fairness in IEEE 802.11 WLANs," *Pervasive and Mobile Computing*, vol. 8, no. 5, pp. 631–645, 2012 (Chapter 5).
- A. Garcia-Saavedra, P. Serrano, and A. Banchs, "An Energy-efficient Strategy using Distributed Opportunistic Scheduling," *Submitted* (Chapter 5).

Additionally to the above, the following papers with related content have been published during the development of this thesis:

- P. Serrano, A. Garcia-Saavedra, M. Hollick, and A. Banchs, "On the energy efficiency of IEEE 802.11 WLANs," in *Proc. of European Wireless (EW)*, 2010, pp. 932–939.
- D. Camps-Mur, A. Garcia-Saavedra, and P. Serrano, "Device to device communications with WiFi Direct: overview and experimentation," *IEEE Wireless Communications Magazine*, Accepted.
- A. Garcia-Saavedra, B. Rengarajan, P. Serrano, D. Camps-Mur, and X. Pérez-Costa, "SOLOR: Self-Optimizing WLANs with Legacy-Friendly Opportunistic Relays," *Submitted*.
- A. Banchs, J. Ortin, A. Garcia-Saavedra, D. J. Leith, and P. Serrano, "Thwarting Selfish Behavior in 802.11e WLANs," *Submitted*.

Chapter 2

Related Work

The pseudorandom oscillations of the wireless channels' signal power to noise power ratio, i.e,. fading effects, are a phenomenon that has been traditionally seen as a source of unreliability that has to be mitigated. With this objective, physical layer (PHY) mechanisms have typically been designed to provide wireless links with larger robustness and reliability, e.g., through proper selection of a coding and modulation scheme. However, fading effects still affect the MAC layer, as well, in many realistic scenarios [4, 44]; and implementing conservative PHY modulation and coding schemes underutilizes the scarce wireless resources [45].

2.1 Opportunistic Scheduling

There is recent surge of interest in channel-aware policies that see fading as a source of *randomization* that can be *exploited*. The key idea of these mechanisms is to leverage multiuser diversity by scheduling transmission opportunities to users only when they experience peaks in the quality of their links, thus increasing the overall performance. Multiuser diversity has its roots in the work of Knopp et al. [46]; in this work, they propose a power control mechanism by which, given the channel gains of each user, the policy that maximizes capacity is the one that allows only the user with best channel to transmit at any time. Multiuser diversity is a diversity technique that underlies a large set of work during the last decade [46–56]. [47, 48] present a scheduling scheme for the Qualcomm/HDR system that exploits time-varying conditions under a proportional fairness (PF) criteria [16]. [49] designs a PF scheduler, as well, with a more flexible structure that allows the system to explicitly set the time fractions for each user and, moreover, outperforms the HDR scheduler in terms of overall throughput. For the EDGE/GPRS system, the possibility of trading off efficiency for fairness when exploiting temporary fluctuations in channel conditions is studied in [50]. [51] provides an opportunistic scheduler

with Quality-of-Service (QoS) guarantees, named M-LWDF (Modified Largest Weighted Delay First), by ensuring a minimum throughput guarantee and also to maintain delays smaller than a predefined threshold value with a given probability for each user. Under a different perspective, some authors propose techniques such as opportunistic beamforming increase the multiuser diversity [52, 53] and extend the performance gains. In [54], an online algorithm (the exponential rule) is designed guaranteeing stability in user's queues without explicit knowledge of arrival rates or channel statics. An opportunistic scheduler under an utility-based fairness criterion is further designed in [55].

2.1.1 Centralized mechanisms

A recent survey on the topic [57] classifies this literature into four categories: capacity, QoS, fairness and distributed scheduling.

Capacity. There are a lot of works proposing opportunistic scheduling schemes that seek to maximize the wireless capacity in multi-user networks. Some of these works [58–61] assume that full Channel State Information (CSI) is available at the base station. Some other works [62–66] consider that the base station does not have instantaneous access to the state information on every link, but instead it estimates this data by e.g., periodic (and possibly discretized, and thus imperfect) reports from mobile users. Cognitive radio networks, where a "secondary" user enjoys of transmission opportunities as long as it minimizes (or removes) any interference to "primary users" have also proposed opportunistic schedulers to increase their performance with both perfect channel information [61] and imperfect information [62].

QoS. Many other works schedule the wireless resources attending to some sort of Quality-of-Service guarantees, such as delay, jitter or minimum throughput. [13] investigated the interaction of QoS and Best Effort (BE) traffic for the first time, and proposed an opportunistic scheduler with throughput guarantees for the QoS flow. In [67], they proposed a "Log rule" scheduling policy which maximizes throughput while guaranteeing certain delay performance. [14] also provides delay guarantees by developing a novel virtual queue technique which guarantees a worst case delay for each user. The work of [68] maximizes system throughput while meeting the required average transmission rate and average absolute deviation of transmission rate to address the scheduling of both real time and BE traffic. [15] sets throughput, delay and packet drops as a multiple QoS objective and propose three schedulers, namely, biggest QoS-deviation first (BQDF), adaptive QoS-deviation control (AQDC), and adaptive residual time control (ARTC).

Fairness. Fairness is typically an issue for opportunistic schedulers. Note that, when preferentially scheduling transmission opportunities to stations with best link conditions, users with lower average conditions could suffer throughput starvation; thus, fairness

criteria are considered in many works in the literature. [69] proposes an opportunistic scheduler for OFDM systems under three different fairness criteria, namely temporal fairness, where all users are guaranteed certain airtime; utilitarian fairness, where they are guaranteed a certain utility of the throughput; and minimum-performance guarantees, where system's performance is maximized while satisfying minimum user requirements. The work carried out in [70] proposes a Proportional Fair (PF) scheduler and [71] introduces four algorithms to exploit user and channel diversity in a PF-based context. The authors of [72] study an adaptive resource allocation mechanism for OFDM systems that takes into account per-user's requirements as a fairness measure. [73] exploits opportunism in TDMA networks and investigate a probabilistic utility-based scheme to account for both elastic (e.g., HTTP, FTP) and non-elastic (e.g., video streaming) services.

2.1.2 Distributed scheduling

The proposals mentioned above target a centralized approach, e.g., with centralized base stations operating a cellular network. In such approaches, the scheduler concentrates all user's channel condition information, either explicitly or through estimation, to take scheduling decisions. Distributed approaches have been proposed only recently. The seminal work of Zheng et al. [7] lays the basic foundations for Distributed Opportunistic Scheduling (DOS) mechanisms suited for wireless distributed networks like ad-hoc communications. The work in [7] studies DOS with a random access protocol in ad-hoc networks where time is divided into slots of duration τ. In DOS, users contend for channel access; if two or more stations attempt to access the channel in a given slot their transmissions collide, if nobody access the channel that slot remains idle or empty and when only one station attempts to access the channel, i.e., a successful contention, it measures its local channel conditions in the first slot. Based on the outcome of this channel probing, the station decides whether to transmit (if the channel conditions are good) for a fixed duration $\mathcal{T}$ or give up (if the channel conditions are bad) its transmission opportunity, leaving the channel free for recontention. Using optimal stopping theory, they show that the optimal scheme is a pure threshold policy, i.e., a station will give up its transmission opportunity (granted by channel contention) if the instantaneous link condition is below a certain threshold.

They first treat DOS as a cooperative game where all users collaborate to maximize overall performance and find the optimal stopping rule as a threshold policy $\bar{R}$ that represents the maximum throughput, and is the unique solution for this fixed point equation:

$$E\left[R(\theta) - \bar{R}^*\right]^+ = \frac{\bar{R}^*\tau}{p_s\mathcal{T}}$$

where $R(\theta)$ is the available transmission rate at time θ, p_s is the probability of a successful

slot, τ is the duration of a slot, $\mathcal{T}$ is the fixed duration of a transmission and $\bar{R}^*$ is the optimal threshold (see Chapter 3 for a more detailed description of the system). They also propose an iterative algorithm to find a threshold $\bar{R}^*$, common to all stations, for scenarios with heterogeneous links (i.e., links with different average conditions) to maximize overall throughput performance. [7] also describes a non-cooperative game, where each user target to maximize its individual throughput and thus, they do not cooperate to benefit the common welfare[1]. The authors propose an online algorithm to configure the threshold that leads to a Nash equilibrium [74] which however, is substantially less efficient than the performance obtained with the cooperative approach. In [10], the authors extend their work in [7] to account for average delay constraints with both network-wide and per-user average constraints, in [9] they extend it to account for imperfect channel state information, and in [8] they improve the channel estimation through a two-level channel probing.

These works rely on the assumption that every channel probing takes an independent observation of the channel while, in reality, there exists some correlation between two adjacent channel observations. This correlation is lower the larger the temporal separation between the observations. [7] justifies this assumption by showing that, for networks with a large number of nodes, the probability of a successful contention (and thus the probability of probing the channel) by station i is very low and the correlation between the observations is therefore very low as well. Very recently, [12] perform a similar work as [7] but they remove this assumption by including in the computation of the threshold the concept of effective observation points. However, these contributions penalize stations with poorer average channel conditions (to achieve overall throughput maximization), causing an unfair allocation of resources which is deeper the greater the heterogeneity of the scenario. Moreover, the cooperative strategy, in turn, requires global knowledge of the channel distribution of all stations, thus, it requires explicit information exchange or its estimation (i.e., cooperation), and the non-cooperative strategy is substantially penalized in terms of overall performance.

In Chapter 3 of this thesis, we solve these issues by proposing an adaptive algorithm which also works well under non-saturation conditions (differently from previous approaches) and drives the system to an optimal Proportional Fair (PF) allocation with the only requirement of local information. Furthermore, in Chapter 4 we design an online algorithm that, in non-cooperative scenarios, drives the Nash equilibrium to the optimal configuration computed in Chapter 3, i.e., without performance losses.

[1] where an overall, rather than individual, performance metric is maximized.

2.2 Energy consumption of wireless devices

The optimization of energy-based criteria require reliable models that accurately captures the power consumption behavior of wireless devices. In this thesis, in order to exploit DOS techniques to perform such optimization, we unveil a power consumption model including some meaningful factors apparently not accounted for in previous literature.

2.2.1 Energy consumption of devices

A number of previous works in the area analyze, like us, the consumption of the complete device, either a laptop [75–77] or a mobile phone [78, 79]. Some of these works deal with specific issues, such as quantification of the consumption of components other than wireless interfaces (e.g., CPU, screen, memory) [78], power consumption measurements via available APIs for estimating the battery discharge state [77], assessment of tradeoff between CPU consumption due to data compression and wireless consumption due to data transmission [76], but do not tackle the per-frame energy consumption domain. Only [75] briefly mentions that the energy consumption associated to packet processing might be non negligible, but does not provide any measurement or evidence. [79] finds that message size can have a non-intuitive impact on the energy consumption, but their guess is either the existence of some power management threshold or a bug in the wireless firmware (indeed, energy bugs in mobile devices are a current concern [80]). We distinguish from all these works in the fact that *we perform a fine-grained per-packet energy consumption decomposition*, versus their energy consumption analyses on a much coarser scale.

2.2.2 Energy consumption of interfaces

Unlike the previous papers, most characterizations of the wireless interface consumption are done on a per-packet basis. The seminal work of [81] shows that transmission/reception of a frame has a linear dependency on its length. This result is caused by the four different states a wireless NIC can be in, namely: sleep, idle, receiving and transmitting. The results are extended in [82] for different modulation and coding schemes and transmission power configurations. While in these cases the wireless interface is treated as a whole, [83] distinguishes between the (approximately constant) Application-Specific Integrated Circuit (ASIC) consumption, and the Power Amplifier (PA) consumption occurring only outside idle periods. None of these works analyze the energy consumption of a frame as it is delivered to/from the NIC.

2.2.3 Energy consumption models

The (implicit or explicit) assumption of all previous energy consumption models [84–86,34,39,87–90] is that the PA operations dominate the consumption of the whole device, which allows to model consumption with a finite number of states, e.g., {active, idle} [84, 89], {transmission, reception, idle} [86, 87, 39], and so on. More specifically, the common approach followed by all these papers (as well as that recently included in the NS3 network simulator [91]) is to model the NIC consumption using data sheet parameters [92], and add to this a fixed amount to account for the non-wireless power consumption of the device. In [90], the authors propose an extended model that accounts for the power conversion efficiency of the PA, but eventually the model suffers from the same limitations. As shown in [41, 42], however, these energy consumption models *fail to capture crucial aspects of how energy is consumed in real world devices*, and therefore their use might bias conclusions.

Chapter 3

Optimizing Throughput Performance and Fairness in DOS

Opportunistic techniques can be used in wireless networks over fading channels to exploit multi-user and time diversity to obtain throughput gains. In this chapter, we first introduce the system model of a network running DOS and describe the throughput performance as a function of the configuration parameters. Then, we find the optimal static configuration of the network to achieve a proportional fair allocation of the individual throughputs and, finally, we design and evaluate an adaptive and distributed algorithm named ADOS ("Adaptive DOS") that drives the system to the optimal point of operation.

3.1 System and throughput model

Our system model follows that of [7–10]. We consider a single-hop wireless network with N stations, where time is divided in mini slots and station i contends for the channel in each mini slot with an access probability p_i. We assume a collision model where a mini slot contains a collision when two or more stations contend, it contains a successful contention when only one station contends and it is empty (or idle) otherwise.

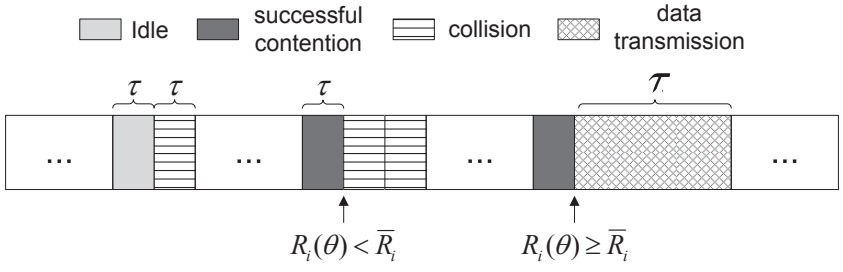


Figure 3.1: Example of channel contention.

We let τ denote the duration of a mini slot. After an empty mini slot or a collision, stations recontend for channel access in the next mini slot. Following a successful contention of station i, the station may transmit depending on the channel conditions. Let $R_i(\theta)$ denote the channel transmission rate of station i at time θ, as given by the capacity of the current channel. If $R_i(\theta)$ is small (indicating a poor channel), station i gives up on this transmission opportunity and lets all the stations recontend in the next mini slot. Otherwise, it transmits after the mini slot containing the successful contention for a duration of $\mathcal{T}$. Fig. 3.1 depicts an example of such channel contention.

3.1.1 Throughput model

Our throughput model, like that of [7–10], assumes that $R_i(\theta)$ remains constant for the duration of a data transmission and that different observations of $R_i(\theta)$ are independent.[1] On the one hand the assumption that $R_i(\theta)$ remains constant during a data transmission is a standard assumption for the block-fading channel in wireless communications [93,94]. On the other hand the assumption that different observations are independent is justified in [7] through numerical calculations which show that in many practical scenarios the channel correlation between two adjacent successful contentions of a station is very small with a very high probability.

Let l_i be the average number of bits that station i transmits following a successful contention and T_i be the average time it holds the channel (including the time spent in contention). Then, the throughput of station i is

$$r_i = \frac{p_{s,i} l_i}{\sum_j p_{s,j} T_j + (1 - p_s)\tau} \tag{3.1}$$

where $p_{s,i}$ is the probability that a mini slot contains a successful contention of station i,

$$p_{s,i} = p_i \prod_{j \neq i} (1 - p_j) \tag{3.2}$$

and p_s is the probability that the minislot contains a successful contention of any station in the system

$$p_s = \sum_i p_{s,i}. \tag{3.3}$$

Both l_i and T_i depend on $\bar{R}_i$. When a station contends successfully, it holds the channel for a time $\mathcal{T} + \tau$ if it transmits data and τ if it gives up the transmission opportunity. Thus, T_i can be computed as

$$T_i = Prob(R_i(\theta) < \bar{R}_i)\tau + Prob(R_i(\theta) \geq \bar{R}_i)(\mathcal{T} + \tau). \tag{3.4}$$

[1] This assumption can be relaxed following the work of [12].

When the station uses the transmission opportunity, it transmits a number of bits given by $R_i(\theta)\mathcal{T}$, which yields

$$l_i = \int_{\bar{R}_i}^{\infty} r\mathcal{T} f_{R_i}(r)dr \tag{3.5}$$

where $f_{R_i}(r)$ is the pdf of $R_i(\theta)$.

With the above, we can describe the throughput of the stations as a function of the configuration parameters of the system, i.e., the vector $\mathbf{p} = \{p_1, \ldots, p_N\}$ and the vector $\bar{\mathbf{R}} = \{\bar{R}_1, \ldots, \bar{R}_N\}$.

3.1.2 Validation

In order to validate our analytical model we set up a scenario with $N = 10$ stations with homogeneous links, i.e., all links have the same average channel quality. We fix each station's access probability to $p_i = 0.1$ and transmission rate threshold to $\bar{R}_i = \bar{R}$ $\forall i \in \{1...N\}$, and compute the aggregated throughput performance for different values of $\bar{R}$ and various channel conditions. Finally, we set up the same topology an run simulations in a Jake's channel simulator[2] [95] that does not assume independent observations but there exist temporal correlation in every observation of the channel instead. We compare the total throughput performance of each experiment as a function of the threshold for scenarios with different average normalized signal to noise ratio (ρ) in Fig 3.2.

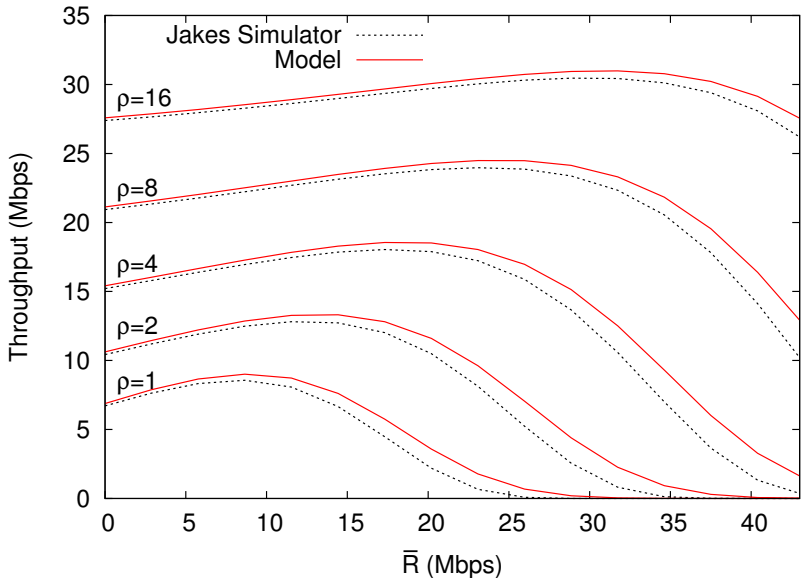


Figure 3.2: Throughput model validation

[2]Results are given for a Doppler frequency of $f_D = 2\pi/100\tau$.

We observe that the results from the throughput model approximates those given by the channel simulator, with a throughput estimation error in the optimal point of operation that spans from 1.5% (for $\rho = 16$) to 5% (for $\rho = 1$). It is also worth noting that the analytical model captures very closely the trends of the simulations and that the optimal point of operation is very similar in both experiments. Given these results, we conclude that the throughput model proposed is adequate to the task of finding the optimal configuration for any optimization strategy we pursue (maximizing total throughput in this example).

3.2 Optimal p_i configuration

We build upon the throughput performance model proposed in Section 3.1, where

$$r_i = \frac{p_{s,i} l_i}{\sum_j p_{s,j} T_j + (1 - p_s)\tau} \tag{3.6}$$

Let us start by computing the optimal configuration of the $\mathbf{p}$ parameters. We define w_i as the ration between the probability of a successful contention by station i and the successful probability of a reference station, namely station 1.

$$w_i = \frac{p_{s,i}}{p_{s,1}} \tag{3.7}$$

From the above equation, we have that $p_{s,i} = w_i p_s / \sum_j w_j$; substituting this into (3.1) yields

$$r_i = \frac{w_i p_s l_i}{\sum_j w_j p_s T_j + \sum_j w_j (1 - p_s)\tau}$$

In a slotted wireless system such as the one of this thesis, the optimal access probabilities satisfy $\sum_i p_i = 1$ (see [96]), which gives an optimal success probability p_s approximately equal to $1/e$,

$$p_s = \sum_i p_i \prod_{j \neq i} 1 - p_j \approx \sum_i p_i e^{-\sum_j p_j} = e^{-1} \tag{3.8}$$

With the above, the problem of finding the $\mathbf{p}$ configuration that maximizes the proportionally fair rate allocation is thus equivalent to finding the w_i values that maximize $\sum_i \log(r_i)$, given that $p_s = 1/e$. To obtain these w_i values, we impose

$$\frac{\partial \sum_i \log(r_i)}{\partial w_i} = 0$$

which yields

$$\frac{1}{w_i} - N\frac{p_s T_i + (1-p_s)\tau}{\sum_i w_i p_s T_i + \sum_j w_j (1-p_s)\tau} = 0$$

Combining this expression for w_i and w_j, we obtain

$$\frac{w_i}{w_j} = \frac{p_s T_j + (1-p_s)\tau}{p_s T_i + (1-p_s)\tau}$$

Given that $w_i/w_j \approx p_i/p_j$ and $p_s = 1/e$, the above can be rewritten as

$$\frac{p_i}{p_j} = \frac{T_j + (e-1)\tau}{T_i + (e-1)\tau} \tag{3.9}$$

Furthermore, the probability that a given mini slot is empty can be computed as follows,

$$p_e = \prod_i 1 - p_i \approx e^{-\sum_i p_i} = e^{-1} \tag{3.10}$$

With the above, we compute the solution of the optimization problem by finding the $\mathbf{p}$ values that solve the system of equations formed by (3.9) and (3.10).

The uniqueness of the solution of this system of equations can be proved as follows. Without loss of generality, let us take the access probability of station 1, p_1, as reference. From (3.9) we have that p_i for $i \neq 1$ can be expressed as a continuous and monotone increasing function of p_1. Applying this to (3.10), we have that the term $(\prod_i 1 - p_i)$ is a continuous and monotone decreasing function of p_1 that starts at 1 and decreases to 0, while the right hand side is the constant value $1/e$. From this, it follows that there is a unique value of p_1 that satisfies this equation. Taking the resulting p_1 and computing $p_i \; \forall i \neq 1$ from (3.9), we have a solution to the system. Uniqueness of the solution is given by the fact that all relationships are bijective and any solution must satisfy (3.10), which (as we have shown) has only one solution.

Hereafter, we denote the unique solution to the system of equations by $\mathbf{p}^* = \{p_1^*, \ldots, p_N^*\}$. Note that determining $\mathbf{p}^*$ requires computing $T_i \; \forall i$, which depend on the optimal configuration of the thresholds $\bar{\mathbf{R}}$. In the following section we address the computation of the optimal $\bar{\mathbf{R}}$, which we denote by $\bar{\mathbf{R}}^* = \{\bar{R}_1^*, \ldots, \bar{R}_N^*\}$.

3.3 Optimal $\bar{R}_i$ configuration

From [7], we have that the optimal transmission policy for our system is a threshold policy: given a threshold $\bar{R}_i$, station i only transmits after a successful contention if $R_i(\theta) \geq \bar{R}_i$.

In order to obtain the optimal configuration of $\bar{\mathbf{R}}$, we need to find the transmission threshold of each station that, given the $\mathbf{p}^*$ computed above, optimizes the overall performance in terms of proportional fairness. This is given by the following theorem.

Theorem 1. *Consider a station k that is alone in the network and contends for the channel with $p_k = 1/e$. Let $\bar{R}_k^1$ be the transmission rate threshold that optimizes the throughput of this station under the assumption that different channel observations are independent. Then, $\bar{R}_k^* = \bar{R}_k^1$.*

Proof. The proof is by contradiction. Assume there exists a configuration $\bar{\mathbf{R}}^*$ with $\bar{R}_k^* \neq \bar{R}_k^1$ for some station k that provides proportional fairness.

Let l_k^1 and T_k^1 be the values of l_k and T_k for the threshold $\bar{R}_k^1$ and l_k^* and T_k^* the corresponding values for $\bar{R}_k^*$. Since $\bar{R}_k^1$ maximizes r_k when station k is alone:

$$\frac{l_k^1}{T_k^1 + (e-1)\tau} > \frac{l_k^*}{T_k^* + (e-1)\tau}. \tag{3.11}$$

Consider a network with N stations that use configuration $\bar{\mathbf{R}}^*$. Given $\bar{\mathbf{R}}^*$, the $\mathbf{p}^*$ that maximizes $\sum_i \log(r_i)$ is given by (3.9) and (3.10). This results in the following throughput for station k:

$$r_k^* = \frac{p_{s,k}^* l_k^*}{\sum_j p_{s,j}^* (T_j^* + (e-1)\tau)} = \frac{l_k^*}{N(T_k^* + (e-1)\tau)} \tag{3.12}$$

and for the other stations:

$$r_i^* = \frac{l_i^*}{N(T_i^* + (e-1)\tau)} \qquad \forall i \neq k. \tag{3.13}$$

Let us now consider the alternative configuration $\bar{R}_k^1$ for station k and $\bar{R}_i^*$ for the other stations. If we take the p_k^1 and p_i^1 configuration that satisfies (3.9) and (3.10) with this alternative configuration, we obtain the following throughput for station k:

$$r_k^1 = \frac{l_k^1}{N(T_k^1 + (e-1)\tau)} > r_k^* \tag{3.14}$$

and for the other stations:

$$r_i^* = \frac{l_i^*}{N(T_i^* + (e-1)\tau)} \qquad \forall i \neq k. \tag{3.15}$$

With the above, we have found an alternative configuration that provides a higher throughput to station k and the same throughput to all other stations. This alternative configuration thus increases $\sum_i \log(r_i)$, which contradicts the initial assumption that the configuration $\bar{\mathbf{R}}^*$ provides proportional fairness. □

Following the above result, the optimal configuration of the threshold $\bar{R}_i^*$ can be obtained by computing the transmission rate threshold $\bar{R}_k^1$ that optimizes the throughput of station k when it is alone in the channel and it contends for the channel with $p_k = 1/e$. This is done in [7], which uses *optimal stopping theory* and finds that the optimal threshold can be obtained by solving the following fixed point equation:

$$E\left[R_i(\theta) - \bar{R}_i^*\right]^+ = \frac{\bar{R}_i^* \tau}{\mathcal{T}/e} \tag{3.16}$$

Note that the above allows computing the threshold $\bar{R}_i^*$ of a station based on *local information* only, as (3.16) does not depend on the other stations in the network and their radio conditions. In particular, the optimal threshold configuration is *independent of the access probabilities* $\mathbf{p}$, which is crucial as it allows decoupling the algorithm that adjusts the configuration of $\bar{\mathbf{R}}$ from the one that adjusts $\mathbf{p}$. In the following, we present two independent adaptive algorithms to bring the system to the optimal point of operation: one that drives the access probabilities $\mathbf{p}$ to their optimal values $\mathbf{p}^*$, and another that drives the threshold of each station $\bar{R}_i$ to its optimal value $\bar{R}_i^*$.

3.4 ADOS Algorithm

In this section, we present the ADOS mechanism, which consists of two independent adaptive algorithms. The first algorithm determines the access probability used by a station, p_i, adjusting the value when the number of active stations in the network or their sending behavior change. The second algorithm determines the transmission rate threshold of a station, $\bar{R}_i$, adapting its value to the changing radio conditions of the station. Both algorithms together drive the system to the optimal point of operation.

While the optimal configuration devised above assumes saturation conditions, the ADOS mechanism applies to saturated and non-saturated wireless networks. In the following, we first discuss the implications of non-saturated stations and then present the two adaptive algorithms. One of the key features of these algorithms is that they do not require to know the number of stations in the network, and they do not need to keep track of the behavior of the other stations or their channel conditions.

3.4.1 Non-saturation conditions

The optimal configuration $\{\mathbf{p}^*, \bar{\mathbf{R}}^*\}$ obtained in the previous section corresponds to the case where all stations are saturated. We next discuss how to consider the case when some of the stations are not saturated.

In the previous section we have seen that, when all the stations are saturated, the optimal channel empty probability p_e takes a constant value equal to $1/e$, independent of the number of stations. The first key approximation is to assume that this also holds when some of the stations are not saturated. The rationale behind this assumption is that when some of the stations are not saturated, they transmit with a smaller access probability, and therefore the other stations should transmit with a higher access probability to achieve optimal throughput efficiency, as otherwise channel time is wasted with empty mini slots. By keeping the same target p_e, we increase the access probabilities of the other stations, and thus adapt their behavior to the actual load in the network.

We have also seen in the previous section that, under saturation, the optimal transmission rate thresholds are constant values that only depend on the local radio conditions. The second key approximation is to assume that the optimal transmission rate thresholds take the same constant values under non-saturation. Note that, according to the analysis of the previous section, these thresholds only depend on the time wasted in contention. As we adjust the access probabilities to the actual load, we can assume that the time wasted in contention is the same as in the saturated case, which leads to the same optimal configuration for the thresholds.

We next present the design of the algorithms to adjust p_i and $\bar{R}_i$ that consider both saturation and non-saturation conditions following the two approximations exposed above.

3.4.2 Adaptive algorithm for p_i

Following the first approximation above, with ADOS each station implements an adaptive algorithm to configure the access probability p_i, with the goal of driving the channel empty probability to $1/e$, as given by (3.8).

Driving the channel empty probability toward a constant optimum value fits well with the framework of *classic control theory*. With these techniques, we measure the *output signal* of the system and, by judiciously adjusting the *control signal*, we aim at driving it to the *reference signal*. A key advantage of using such techniques is that they provide the means for achieving a good tradeoff between the speed of reaction and stability while guaranteeing convergence, which is a major challenge when designing adaptive algorithms.

Fig. 3.3 depicts our algorithm to adjust $\mathbf{p}$, where each station computes the error signal E_p by subtracting the output signal O_p from the reference signal R_p (the functions in the figure are given in the z domain). The output signal O_p is combined with a noise component W_p of zero mean, modeling the randomness of the channel access algorithm. In order to eliminate this noise, we follow the design guidelines from [97] and introduce a low-pass filter $F_p(z)$. The filtered error signal $\hat{E}_p$ is then fed into the controller $C_{p,i}(z)$ of each station, which provides the control signal t_i, defined as the average time between

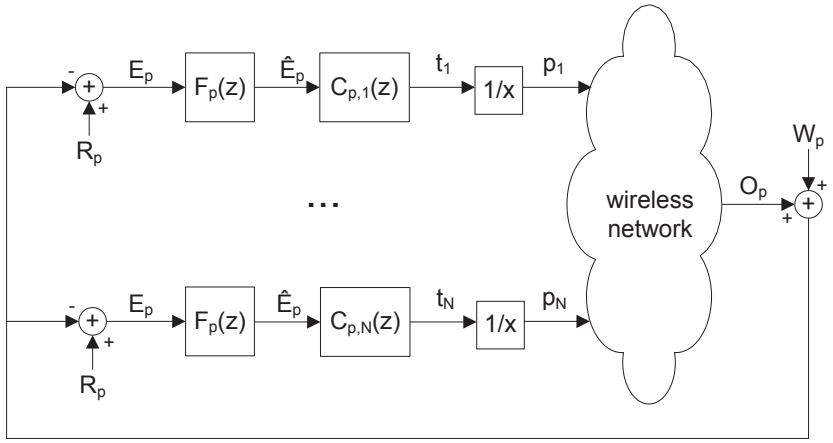


Figure 3.3: Adaptive algorithm for p_i.

two transmissions of station i. Station i then computes its access probability as $p_i = 1/t_i$. With the p_i of each station, the wireless network provides the output signal O_p, which closes the loop.

In the above system, we need to design the reference and output signals R_p and O_p, as well as the transfer functions of the low-pass filter and the controller, $F_p(z)$ and $C_{p,i}(z)$. In the following we address the design of these components with the goal of ensuring that the empty probability p_e is driven to $1/e$.

In our system, time is divided into intervals such that the end of an interval corresponds to a transmission in the channel (either a success or a collision). Given that the target empty probability is equal to $1/e$, the target average number of empty mini slots between two transmissions (i.e., our reference signal) is equal to $R_p = 1/(e-1)$ in average. In this way, after the n-th transmission, each station computes the output signal at interval n, denoted by $O_p(n)$, as the number of empty mini slots between the $(n-1)$-th and the n-th transmission. The error signal for the next interval is then computed as

$$E_p(n+1) = R_p - O_p(n). \tag{3.17}$$

With the above, if p_e is too large then $O_p(n)$ will be larger than R_p in average, yielding a negative error signal $E_p(n+1)$ that will decrease t_i for the next interval, which will increase the transmission probability p_i and therefore reduce p_e (and vice-versa). This ensures that p_e will be driven to the optimal value.

For the low-pass filter $F_p(z)$, we use a simple exponential smoothing algorithm of parameter α_p [98], given by the following expression in the time domain,

$$\hat{E}_p(n) = \alpha_p E_p(n) + (1-\alpha_p)\hat{E}_p(n-1)$$

which corresponds to the following transfer function in the z domain:

$$F_p(z) = \frac{\alpha_p}{1-(1-\alpha_p)z^{-1}}$$

For the transfer function of the controllers $C_{p,i}(z)$, we use a very simple controller from classical control theory, namely the Proportional Controller [99], which has already been used in a number of networking problems (see e.g. [100, 101]):

$$C_{p,i}(z) = K_{p,i}$$

where $K_{p,i}$ is a per-station constant.

In addition to driving the empty probability to $1/e$, we also impose that the access probabilities satisfy (3.9).[3] Since we feed the same error into the different stations, and the proportional controller simply multiplies this error by a constant to compute p_i, the following equation holds for all i, j:

$$\frac{p_i}{p_j} = \frac{K_{p,j}}{K_{p,i}}$$

Therefore, by simply setting $K_{p,i}$ as

$$K_{p,i} = K_p \left(T_i + (e-1)\tau\right)$$

we ensure that (3.9) is satisfied.

3.4.3 Adaptive algorithm for $\bar{R}_i$

Following the second approximation of Section 3.4.1, the adaptive algorithm of ADOS to adjust the threshold $\bar{R}_i$ aims to drive the threshold of all (saturated and non-saturated) stations to the optimal value given by (3.16). Note that (3.16) is equivalent to the following equation:

$$E\left[(R_i(\theta) - \bar{R}_i^*)^+ - \frac{\bar{R}_i^*\tau}{\mathcal{T}/e}\right] = 0 \tag{3.18}$$

In the following, we design an adaptive algorithm that drives $\bar{R}_i$ to the value given by the above equation. The algorithm is depicted in Fig. 3.4. Similarly to the adaptive algorithm for p_i, we base the algorithm design on control theory. The key difference between the two algorithms is that, since the optimal value of threshold of a station

[3] In Sections 3.2 and 3.3 we have seen that (3.9) needs to be satisfied by the saturated stations. For the non-saturated stations, the transmission probability depends on the sending behavior and not on p_i. Therefore, we can impose (3.9) on all stations and thus avoid differentiating saturated from non-saturated stations.

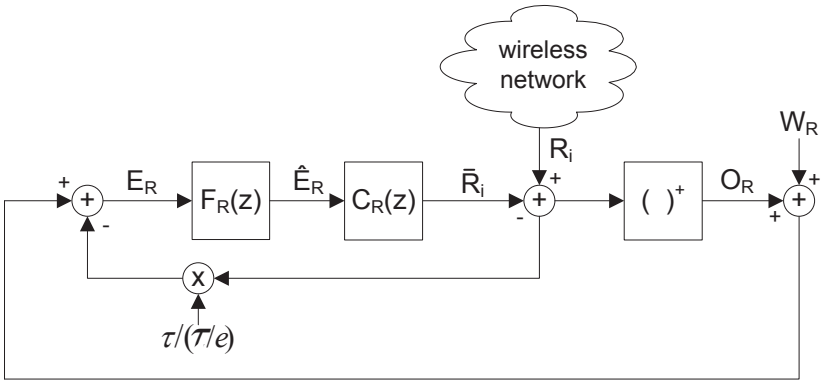


Figure 3.4: Adaptive algorithm for $\bar{R}_i$.

depends on local information only and hence does not depend on the threshold value of the other stations, we can consider each station separately (in contrast to Fig. 3.3).

In order to ensure that the configuration of $\bar{R}_i$ satisfies (3.18), we design the output signal of the algorithm, O_R, equal to the term $(R_i - \bar{R}_i)^+$, and the reference signal, R_R, equal to the term $\bar{R}_i\tau/(\mathcal{T}/e)$. Thus, by driving the difference with these two terms (i.e., the error signal) to zero, we ensure that (3.18) is satisfied.

Following the above, upon its n^{th} successful contention, a station measures the channel transmission rate $R_i(n)$ and computes the output signal as

$$O_R(n) = \begin{cases} R_i(n) - \bar{R}_i(n), & \text{if } R_i(n) >= \bar{R}_i(n) \\ 0, & \text{otherwise} \end{cases}$$

From the above output signal, it then computes the error signal as

$$E_R(n+1) = O_R(n) - \frac{\bar{R}_i(n)\tau}{\mathcal{T}/e}$$

Due to the randomness of the radio signal, the output signal carries some noise W_R. In order to filter out this noise, we apply (like in the previous case) a low pass-filter $F_R(z)$ on the error signal, which yields

$$\hat{E}_R(n) = \alpha_R E(n) + (1 - \alpha_R)\hat{E}_R(n-1)$$

Also like in the previous case, the error signal is introduced into a proportional controller,

$$C_R(z) = K_R$$

where K_R is the constant of the controller.

The controller gives the threshold configuration $\bar{R}_i(n)$ as output. As mentioned above, by driving the error signal $\hat{E}_R(n)$ to 0, the controller ensures the threshold value satisfies (3.18) and thus achieves the objective of adjusting the treshold to the optimal value $\bar{R}_i^*$ obtained in Sections 3.2 and 3.3.

3.5 Control Theoretic Analysis

With the above, we have all the components of the ADOS mechanism fully designed. The remaining challenge is the setting of its parameters, that is, the parameters of the adaptive algorithm for p_i (K_p and α_p) and the adaptive algorithm for $\bar{R}_i$ (K_R and α_R). In this section, we conduct a control theoretic analysis of the algorithms to find a suitable parameter setting.

As discussed in Sections 3.2 and 3.3, the setting of the optimal threshold $\bar{R}_i^*$ does not depend on the configuration of $\mathbf{p}$. Based on this, we analyze the closed-loop behavior of the two adaptive algorithms independently. For the adaptive algorithm to adjust $\bar{R}_i$, the behavior is independent of the $\mathbf{p}$ configuration. For the algorithm to adjust p_i, we consider that the values of $\bar{\mathbf{R}}$ are fixed, as their configuration depends only on the radio conditions, and analyze the convergence of p_i to the optimal configuration corresponding to these $\bar{\mathbf{R}}$ values.

In the following, we first analyze the adaptive algorithm to adjust p_i and then we analyze the one to adjust $\bar{R}_i$; as a result, we configure the parameters of the algorithms in such a way that a good tradeoff between stability and speed of convergence is achieved.

3.5.1 Analysis of the algorithm for p_i

We next conduct a control theoretic analysis of the closed-loop system of the algorithm for p_i to find good values for the parameters K_p and α_p. Fig. 3.5 depicts the closed-loop system for this algorithm. Note that the term z^{-1} in the figure shows that the error signal E at a given interval is computed with the output signal O of the previous interval.

In order to analyze this system from a control theoretic standpoint, we need to characterize the transfer function $H_{p,i}$, which takes t_i as input and gives O_p as output. The following equation gives a nonlinear relationship between O_p and $\{t_1, \ldots, t_N\}$:

$$O_p = \frac{1}{1 - p_e} - 1$$

where

$$p_e = \prod_j (1 - 1/t_j)$$

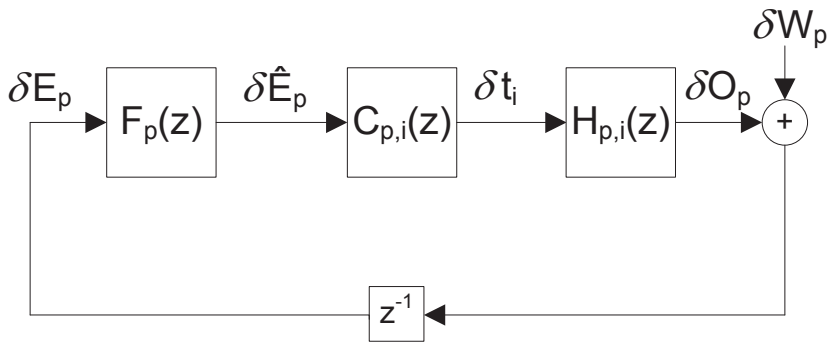


Figure 3.5: Closed-loop system of the adaptive algorithm for p_i.

To express the above relationship as a transfer function, we linearize it when the system suffers small perturbations around its stable point of operation. Then, we study the linearized model and force that it is stable. Note that the stability of the linearized model guarantees that our system is locally stable.[4]

We express the perturbations around the stable point of operation as follows:

$$t_i = t_i^* + \Delta t_i$$

where $t_i^* = 1/p_i^*$ is the stable point of operation of t_i, and Δt_i are the perturbations around this point of operation.

With the above, the perturbations suffered by O_p can be approximated by

$$\Delta O_p = \sum_j \frac{\partial O_p}{\partial t_j} \Delta t_j$$

where

$$\frac{\partial O_p}{\partial t_j} = \frac{\partial O_p}{\partial p_j} \frac{\partial p_j}{\partial t_j} = \frac{p_e\, p_j^2}{(1-p_j)(1-p_e)^2}$$

Given that $t_i/t_j = (T_i + (e-1)\tau)/(T_j + (e-1)\tau)$, the above can be rewritten as

$$\Delta O_p = \left(\sum_j \frac{(T_j + (e-1)\tau) p_e\, p_j^2}{(T_i + (e-1)\tau)(1-p_j)(1-p_e)^2} \right) \Delta t_i$$

With the above, we have characterized $H_{p,i}$:

$$H_{p,i} = \sum_j \frac{(T_j + (e-1)\tau) p_e\, p_j^2}{(T_i + (e-1)\tau)(1-p_j)(1-p_e)^2}$$

[4] A similar approach was used in [102] to analyze RED from a control theoretic standpoint.

The closed-loop transfer function for station i is then given by

$$T_{p,i}(z) = \frac{-z^{-1}C_{p,i}(z)F_p(z)H_{p,i}(z)}{1 + z^{-1}C_{p,i}(z)F_p(z)H_{p,i}(z)}$$

Substituting the expressions for $F_p(z)$, $C_{p,i}(z)$ and $H_{p,i}(z)$ yields

$$T_{p,i}(z) = \frac{-\alpha_p H_{p,i} K_{p,i}}{z - (1 - \alpha_p - \alpha_p K_{p,i} H_{p,i})} \tag{3.19}$$

To guarantee stability, we need to ensure that the zero of the denominator of $T_{p,i}(z)$ falls inside the unit circle $|z| < 1$ [103], which implies

$$K_p < \frac{2 - \alpha_p}{\alpha_p} \frac{1}{\sum_j \frac{(T_j + (e-1)\tau) p_e \, p_j^2}{(1-p_j)(1-p_e)^2}}$$

The problem with the above upper bound is that it depends on the number of stations and their channel conditions. In order to assure stability, we need to obtain an upper bound that guarantees stability independent of these parameters. To do this, we observe that the right hand side of the above inequality takes a minimum value when $N = 1$ and $T_1 = \tau + \mathcal{T}$. Therefore, by setting K_p as follows, we guarantee that the above inequality will be met independent of the number of stations and their channel conditions:

$$K_p < \frac{2 - \alpha_p}{\alpha_p \left(\mathcal{T} + e\tau\right)}$$

The above provides the maximum K_p value that guarantees stability, which we denote by K_p^{max},

$$K_p^{max} = \frac{2 - \alpha_p}{\alpha_p \left(\mathcal{T} + e\tau\right)}$$

In order to set K_p to a value that provides a good tradeoff between the speed of reaction to changes and stability, we follow the Ziegler-Nichols rules [99], which are widely used to configure proportional controllers. According to these rules, this parameter cannot be larger than one half of the maximum value that guarantees stability, which we denote by $K_p^{stability}$:

$$K_p \leq K_p^{stability} = \frac{K_p^{max}}{2} \tag{3.20}$$

In addition to the above, K_p also needs to be set to eliminate the noise from the system. Noise is generated by the randomness of the output signal, which is given by the number of empty mini slots between two transmissions and hence follows a geometric random variable of factor $1 - p_e = 1 - 1/e$. Hence, the noise at the input of the low-pass

filter has a zero mean and a variance given by:

$$E[W_p^2] = \frac{p_e}{(1-p_e)^2} = \frac{1/e}{(1-1/e)^2}$$

The noise at the output of the controller can be obtained from the noise at the input of the low-pass filter with the following transfer function:

$$T_{W_p}(z) = \frac{-z^{-1}C_{p,i}(z)F_p(z)}{1+z^{-1}C_{p,i}(z)F_p(z)H_{p,i}(z)}$$

Substituting $C_{p,i}(z)$, $F_p(z)$ and $H_{p,i}(z)$ into the above yields

$$T_{W_p}(z) = \frac{-z^{-1}\alpha_p K_{p,i}}{1-z^{-1}(1-\alpha_p(1+K_{p,i}H_{p,i}))}$$

With the above transfer function, we can compute the variance of the noise at the output of the controller, denoted by $W_{p,c}$, as follows:

$$E[W_{p,c}^2] = \frac{\alpha_p^2 K_{p,i}^2}{1-(1-\alpha_p(1+K_{p,i}H_{p,i}))^2}\, E[W_p^2]$$

From the above equation, and taking into account from (3.19) and (3.20) that

$$\alpha_p(1+K_{p,i}H_{p,i}) \leq 1+\alpha_p/2$$

we can obtain the following upper bound for $E[W_{p,c}^2]$:

$$E[W_{p,c}^2] \leq \frac{\alpha_p K_{p,i}}{(1-\alpha_p/2)H_{p,i}}\, E[W_p^2]$$

To limit the impact of the noise, we impose a gain factor of at least G_p of the signal level at the output of the controller, $E[S_p^2]$, over the noise level at the same point, $E[W_{p,c}^2]$:

$$\frac{E[S_p^2]}{E[W_{p,c}^2]} \geq G_p$$

The signal at the output of the controller is equal to t_i, which yields $E[S_p^2] = t_i^2$. Combining this with the inequality of (3.21), we have that the following condition is sufficient to provide the desired gain:

$$\frac{t_i^2(1-\alpha_p/2)H_{p,i}}{\alpha_p K_{p,i} E[W_p^2]} \geq G_p$$

Isolating K_p from the above yields

$$K_p \leq \frac{t_i^2(1-\alpha_p/2)}{G_p\alpha_p E[W_p^2]} \sum_j \frac{(T_j+(e-1)\tau)p_e\, p_j^2}{(T_i+(e-1)\tau)^2(1-p_j)(1-p_e)^2}$$

which is satisfied as long as the following condition holds,

$$K_p \leq \frac{1-\alpha_p/2}{G_p\alpha_p} \sum_j \frac{T_j+(e-1)\tau}{(T_i+(e-1)\tau)^2}$$

To find an upper bound that is independent of the number of stations and their conditions, we observe that the right hand side of the above inequality takes a minimum for $N = 1$ and $T_1 = \tau + \mathcal{T}$, which leads to the following upper bound, which we denote by K_p^{noise},

$$K_p \leq K_p^{noise} = \frac{1-\alpha_p/2}{G_p\alpha_p\,(\mathcal{T}+e\tau)}$$

The analysis conducted in this section has given two upper bounds, $K_p^{stability}$ and K_p^{noise}, which guarantee that on the one hand the system is stable and on the other hand the noise level is not excessive. As these bounds depend on α_p and G_p, we also need to find a setting for these parameters. In order to provide a good level of protection against noise, G_p needs to be sufficiently large. Additionally, in order to allow sufficiently large $K_{p,i}$ values, which is needed to avoid a large steady state error at the input of the controllers, $G_p\,\alpha_p$ needs to be sufficiently small. Following these considerations, we set $G_p = 10^2$ and $\alpha_p = 10^{-4}$. With these α_p and G_p values, we then configure K_p as follows:

$$K_p = \min(K_p^{noise}, K_p^{stability})$$

which ensures that the two objectives concerning stability and noise are met.

3.5.2 Analysis of the algorithm for $\bar{R}_i$

We next conduct a control theoretic analysis of the closed-loop system of the algorithm for $\bar{R}_i$, depicted in Fig. 3.6. This analysis follows the same steps as the one above.

The perturbations around the point of equilibrium can be expressed as

$$\bar{R}_i = \bar{R}_i^* + \Delta\bar{R}_i$$

and the perturbations suffered by E_R can be approximated by

$$\Delta E_R = H_R \cdot \Delta\bar{R}_i$$

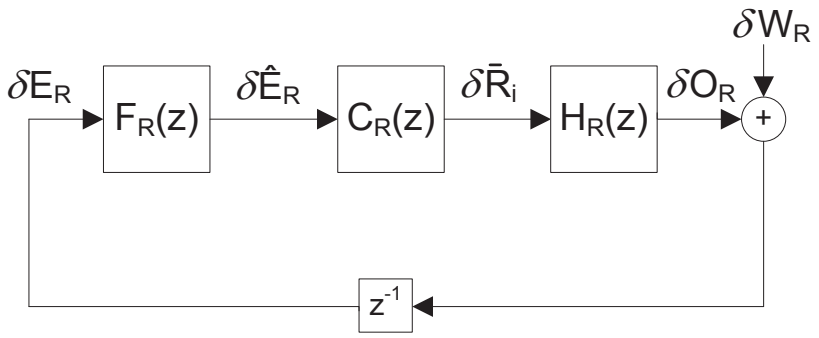


Figure 3.6: Closed-loop system of the adaptive algorithm for $\bar{R}_i$.

where

$$H_R = \frac{\partial E_R}{\partial \bar{R}_i} = \frac{\partial}{\partial \bar{R}_i}\left((R_i - \bar{R}_i)^+ - \frac{\bar{R}_i \tau}{\mathcal{T}/e}\right) = \frac{\partial (R_i - \bar{R}_i)^+}{\partial \bar{R}_i} - \frac{\tau}{\mathcal{T}/e}$$

To compute $\partial(R_i - \bar{R}_i)^+ / \partial \bar{R}_i$, we note that $(R_i - \bar{R}_i)^+$ expresses an average value, as the variations around this average value are captured by another component, namely the noise W_R. For the calculation of the average, we take all possible R_i values weighted by R_i's pdf, $f_{R_i}(r)$, which yields

$$\begin{aligned}\frac{\partial (R_i - \bar{R}_i)^+}{\partial \bar{R}_i} &= \frac{\partial}{\partial \bar{R}_i} \int_{\bar{R}_i}^{\infty} (r - \bar{R}_i) f_{R_i}(r) dr = \\ &= -\int_{\bar{R}_i}^{\infty} f_{R_i}(r) dr\end{aligned}$$

With the above, H_R can be expressed as

$$H_R = -H_{R,1} - H_{R,2}$$

where $H_{R,1} = e\tau/\mathcal{T}$ and $0 \leq H_{R,2} \leq 1$.

The closed-loop transfer function of the system is given by

$$T_R(z) = \frac{C_R(z) F_R(z) H_R(z)}{1 - z^{-1} C_R(z) F_R(z) H_R(z)}$$

where

$$F_R(z) = \frac{\alpha_R}{1 - (1 - \alpha_R) z^{-1}}$$

and

$$C_R(z) = K_R$$

Substituting the expressions for $F_R(z)$, $C_R(z)$ and $H_R(z)$ yields

$$T_R(z) = \frac{-\alpha_R K_R (H_{R,1} + H_{R,2})}{1 - z^{-1}(1 - \alpha_R - K_R \alpha_R (H_{R,1} + H_{R,2}))}$$

To guarantee stability, we need to ensure that the zero of the denominator of $T_R(z)$ falls inside the unit circle $|z| < 1$, which implies

$$K_R < \frac{2 - \alpha_R}{\alpha_R (H_{R,1} + H_{R,2})}$$

In order to find a sufficient condition that holds for all cases, we consider the worst case $H_{R,2} = 1$, which leads to

$$K_R < \frac{2 - \alpha_R}{\alpha_R (1 + e\tau/\mathcal{T})}$$

According to Ziegler-Nichols rules, to guarantee stability we take a K_R value equal to half of the above value,

$$K_R^{stability} = \frac{2 - \alpha_R}{2\alpha_R (1 + e\tau/\mathcal{T})}$$

The noise introduced into the system, W_R, is given by the randomness in the transmission rate values R_i. If we assume that the available transmission rate for a given SNR is given by the Shannon channel capacity, then $R_i = W \log(1 + \rho|h|^2)$, where W is a constant parameter, $\rho|h|^2$ is the SNR and h is the normalized random gain of the channel ($E[h] = 1$). Note that the values of R_i below $\bar{R}_i$ are eliminated from the system by the module that performs the operation $(R_i - \bar{R}_i)^+$, which reduces the noise in the system. In what follows, we do not consider this effect in order to obtain an upper bound on the noise, which provides a worst-case analysis.

If we represent the SNR as the sum of its average value (ρ) plus some noise of zero mean (which we denote by W_h), then we can express the transmission rates R_i as

$$R_i = C \log(1 + \rho + W_h)$$

which we can approximate at the stable point of operation ($W_h = 0$) by

$$R_i \approx C \log(1 + \rho) + W_h \frac{\partial R_i}{\partial W_h}\bigg|_{W_h = 0}$$

Since the noise introduced into the system is given by the variations of R_i around its

average value, from the above we have that we can approximate W_R by

$$W_R \approx W_h \frac{\partial R_i}{\partial W_h}\bigg|_{W_h=0} = \frac{C}{1+\rho} W_h$$

With the above approximation, we can compute the variance of W_R as follows,

$$E[W_R^2] = \frac{C^2}{(1+\rho)^2} E[W_h^2]$$

If we assume that the channel follows a Rayleigh fading model, then $\rho|h|^2$ corresponds to an exponential random variable of rate ρ^{-1}. With this, we have that $E[W_h^2] = \rho^2$, which yields

$$E[W_R^2] = \frac{C^2\rho^2}{(1+\rho)^2}$$

If we denote the noise at the output of the controller by $W_{R,c}$, we have

$$W_{R,c}(z) = \frac{F_R(z)C_R(z)}{1 - z^{-1}F_R(z)C_R(z)H_R(z)} W_R(z)$$

from which

$$W_{R_c}(z) = \frac{\alpha_R K_R}{1 - z^{-1}(1 - \alpha_R - K_R\alpha_R(H_{R,1} + H_{R,2}))} W_R(z)$$

From the above, the variance of the noise at the output of the controller can be computed as

$$E[W_{R,c}^2] = \frac{(\alpha_R K_R)^2}{1 - (1 - \alpha_R(1 + K_R(H_{R,1} + H_{R,2})))^2} E[W_R^2]$$

Given that $K_R \leq K_R^{stability}$, we can obtain the following upper bound on $E[W_{R,c}^2]$:

$$E[W_{R,c}^2] \leq \frac{\alpha_R K_R}{(H_{R,1} + H_{R,2})(1 - \alpha_R/2)} E[W_R^2] \tag{3.21}$$

In order to guarantee a gain of G_R of the signal over the noise at the output of the controller, we impose

$$\frac{E[S_R^2]}{E[W_{R,c}^2]} \geq G_R \tag{3.22}$$

where the signal is the threshold $\bar{R}_i$, which we approximate by the average transmission rate, $C\log(1+\rho)$. With this and the upper bound of (3.21) for $E[W_{R,c}^2]$, we can obtain the following sufficient condition to guarantee (3.22):

$$\left(\frac{\log(1+\rho)(1+\rho)}{\rho}\right)^2 \frac{(H_{R,1} + H_{R,2})(1 - \alpha_R/2)}{\alpha_R K_R} \geq G_R$$

Isolating K_R from the above yields

$$K_R \leq \left(\frac{\log(1+\rho)(1+\rho)}{\rho}\right)^2 \frac{(H_{R,1}+H_{R,2})(1-\alpha_R/2)}{\alpha_R G_R}$$

In order to find a value of K_R that ensures the desired gain for all scenarios, we chose the ρ value that minimizes the right hand side of the above equation and take the worst case value for $H_{R,1}$, which leads to the following upper bound on K_R, which we denote by K_R^{noise},

$$K_R \leq K_R^{noise} = \frac{e\tau(1-\alpha_R/2)}{\mathcal{T}\alpha_R G_R}$$

Similarly to Section 3.5.1, we set $G_R = 10^2$ and $\alpha_R = 10^{-4}$ and choose $K_R = \min(K_R^{noise}, K_R^{stability})$, which ensures that the two goals in terms of noise and stability are met.

3.6 Performance Evaluation

In this section, we present a performance evaluation of ADOS by means of simulations. Unless otherwise stated, we assume that different observations of the channel conditions are independent and that the available transmission rate for a given SNR is given by the Shannon channel capacity:

$$R(h) = W \log_2(1+\rho|h|^2) \text{ bits/s} \tag{3.23}$$

where W is the channel bandwidth in Hz, ρ is the normalized average SNR and h is the random gain of Rayleigh fading.

We implemented the ADOS mechanism in OMNET++.[5] In the simulations, we set $W = 10^7$ and $\mathcal{T}/\tau = 10$. For all results, sufficient number of experiments were executed to achieve that 95% confidence intervals are below 1%.

3.6.1 Homogeneous scenario with saturated stations

We start by considering a homogeneous scenario where all stations are saturated and have the same normalized average SNR ($\rho_i = 1\ \forall i$). We compare the performance of ADOS to the following approaches:

(i) The static optimal configuration obtained from performing an exhaustive search over the $\{p_i, \bar{R}_i\}$ space and choosing the best configuration ('*static configuration*').

[5] `http://www.omnetpp.org/`

(*ii*) An approach that does not perform opportunistic scheduling but always transmits after successful contention ('*non-opportunistic*'), and uses optimal access probabilities.

(*iii*) The team game approach proposed in [7] ('*TDOS*'). This approach requires that each station knows the channel state of all the stations in the network, and hence would incur substantial signaling overhead.

(*iv*) The non-cooperative approach proposed in [7] ('*NDOS*'). This approach, like ours, only requires information that can be observed locally, and hence does not involve any signaling.[6]

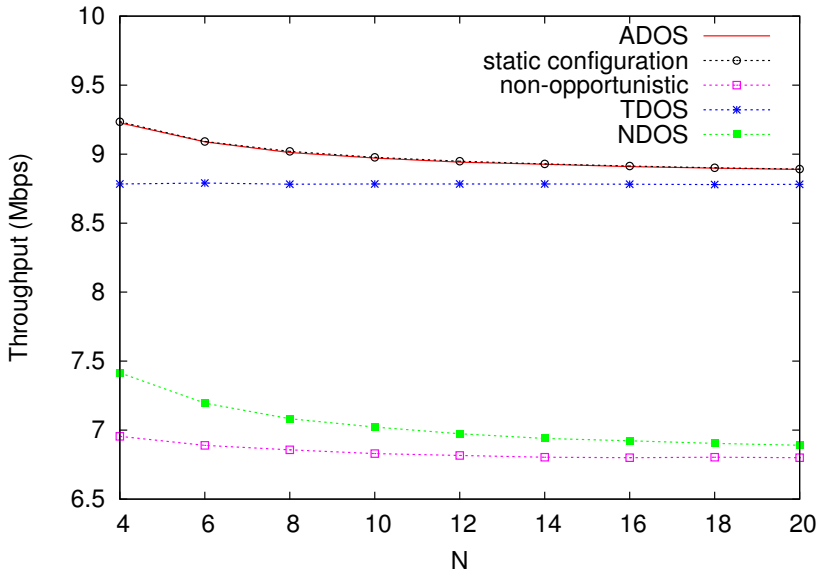


Figure 3.7: Homogeneous scenario with saturated stations.

Fig. 3.7 shows the total throughput as a function of the number of stations in the network. The figure confirms that ADOS is effective in driving the system to the optimal point of operation, providing the same throughput as the benchmark given by the *static configuration*. The *TDOS* and *NDOS* approaches provide lower throughput as they only optimize the transmission rate thresholds; among them, *NDOS* performs substantially worse as it has less information. Finally, the *non-opportunistic* approach provides the lowest throughput due to the lack of opportunistic scheduling. In conclusion, the proposed ADOS mechanism provides optimal throughput performance, outperforming the other approaches.

[6]Since [7] only optimizes the transmission rate thresholds but not the access probabilities, for the '*TDOS*' and '*NDOS*' approaches we take the configuration of access probabilities that are used in the simulation results of [7]. For the '*non-opportunistic*' approach, we choose the access probabilities that maximize the performance, by adapting the analysis of Sections 3.2 and sec:throughput:ri-configuration to the case when stations always transmit after successful contention.

3.6.2 Heterogeneous scenario with saturated stations

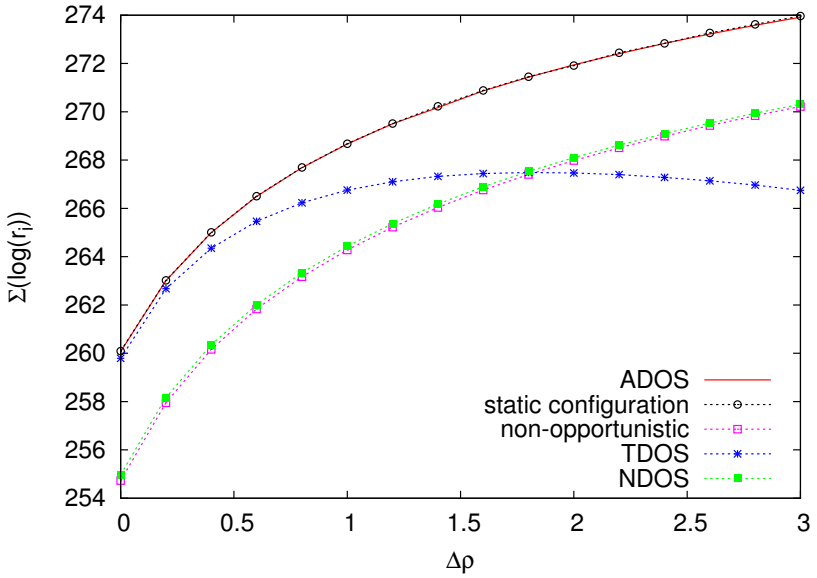


Figure 3.8: Heterogeneous scenario with saturated stations.

In the case of heterogeneous channel conditions, performance does not only depend on the total throughput but also on the way this throughput is shared among the stations. To analyze performance in this scenario, we consider $N = 20$ saturated stations divided into four groups according to their channel conditions. The normalized SNR of the stations from group i is given by $\rho_i = 1 + (i-1)\Delta\rho$, with $i \in \{1, 2, 3, 4\}$. Fig. 3.8 shows $\sum_i \log(r_i)$, the figure of merit for proportional fairness, as a function of $\Delta\rho$. We observe that ADOS performs at the same level as the benchmark given by the *static configuration*, while the other approaches provide a substantially lower performance. *TDOS* exhibits an increasing degree of unfairness as $\Delta\rho$ grows that harms its performance in terms of proportional fairness. *NDOS*, in contrast to *TDOS*, does not show this behavior: with *NDOS*, each station sets its threshold based on its local radio conditions and therefore the fact that other stations have better radio conditions does not impact fairness. The price that *NDOS* pays for this non-cooperative behavior, however, is that the overall throughput performance is substantially degraded for all $\Delta\rho$ values. The *non-opportunistic* approach also provides a poor throughput performance, similar to *NDOS*.

In order to gain additional insight into the throughput distribution with heterogeneous radio conditions, Fig. 3.9 depicts the throughput obtained by a station of each group, for a $\Delta\rho = 2$, with the different approaches, along with the Jain's fairness index (JFI) [104] of each distribution. The results confirm that *TDOS* suffers from high unfairness with heterogeneous radio conditions, since with this approach the stations with worst radio conditions (r_1) are almost starved while the stations with best radio conditions (r_4) obtain

a very large throughput. In contrast, the *TDOS* and *non-opportunistic* approaches do not suffer from unfairness but provide significantly smaller throughputs than ADOS. We conclude that ADOS substantially outperforms all other approaches with heterogeneous radio conditions.

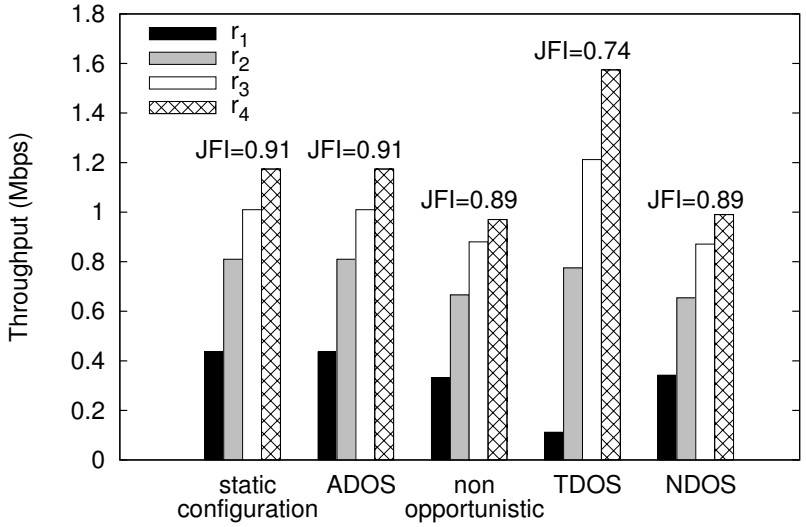


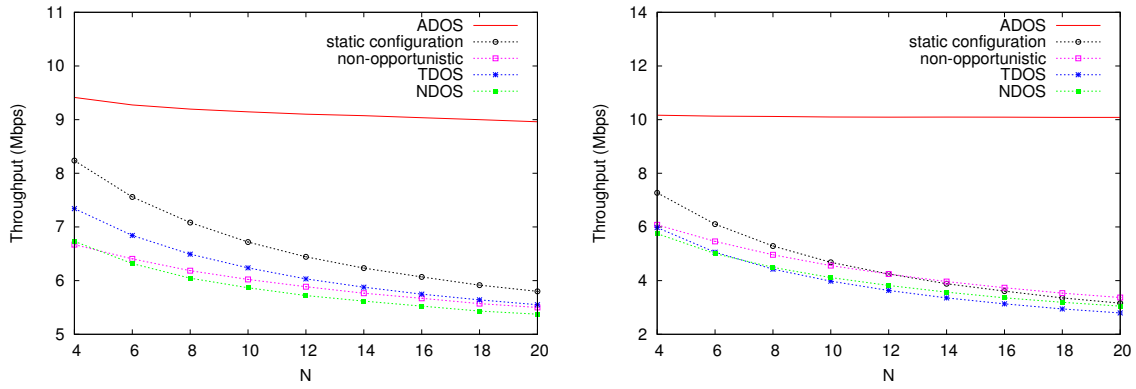
Figure 3.9: Heterogeneous scenario with saturated stations: $\Delta\rho = 2$.

3.6.3 Homogeneous scenario with non-saturated stations

The experiments conducted in previous sections considered saturated conditions, i.e., all stations always had data ready for transmission. In order to assess performance in the case of non-saturation, we consider a scenario with homogeneous radio conditions ($\rho_i = 1\ \forall i$) with one saturated station and $N-1$ non-saturated stations. Figs. 3.10a and 3.10b illustrate the total throughput of the wireless network as a function of the number of stations, when each non-saturated station transmits at one half and one tenth of their saturation throughput (i.e., the throughput the would obtain if they were saturated), respectively. We observe from the figures that ADOS significantly outperforms all other approaches and that this effect becomes more accentuated as the throughput of the non-saturated stations decreases. The reason is that the other approaches assume that all stations are always saturated, and therefore the access probabilities they use become overly conservative for the non-saturated case.

3.6.4 Heterogeneous scenario with non-saturated stations

To evaluate the performance improvement achieved by ADOS with non-saturated stations in the case of a heterogeneous scenario, we repeat the experiment of Fig. 3.8 with



(a) Half saturation rate.

(b) 1/10th saturation rate.

Figure 3.10: Homogeneous scenario with non-saturated stations.

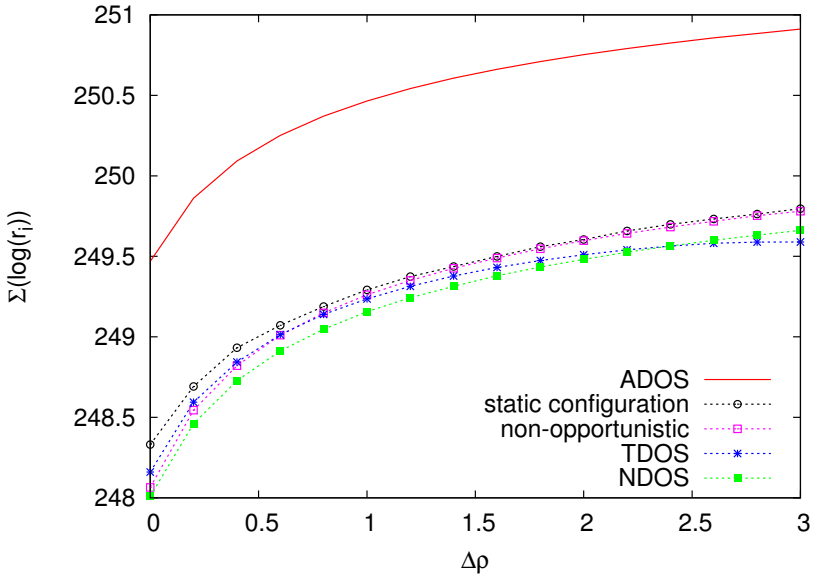


Figure 3.11: Heterogeneous scenario with non-saturated stations.

one of the stations with the highest SNR saturated and the rest of the stations sending at half their saturation throughput. The results, given in Fig. 3.11, show that ADOS also substantially outperforms the other approaches in this case. We further observe that as $\Delta\rho$ grows, the performance of *TDOS* becomes worse than that of the other approaches but does not degrade as in Fig. 3.8. The reasons for this are twofold. On the one hand, there are no fairness issues as only one station is saturated. On the other hand, as $\Delta\rho$ grows, *TDOS* increases the $\bar{R}_i$ of all stations and as a result the non-saturated stations with a small SNR need more successful contentions to send their traffic.

3.6.5 Delay performance

So far, our evaluation has focused on the throughput performance of the wireless network when one or more stations congest the network by transmitting above their saturation throughput. Another relevant performance metric is the average delay introduced when the network is not congested, i.e., when all stations transmit below the saturation throughput and hence are non-saturated. Note that delay is affected, among other aspects, by the transmission opportunities a station gives up before transmitting (which depends on the setting of the threshold) as well as the efficiency of the contention resolution (which depends on the setting of the access probabilities).

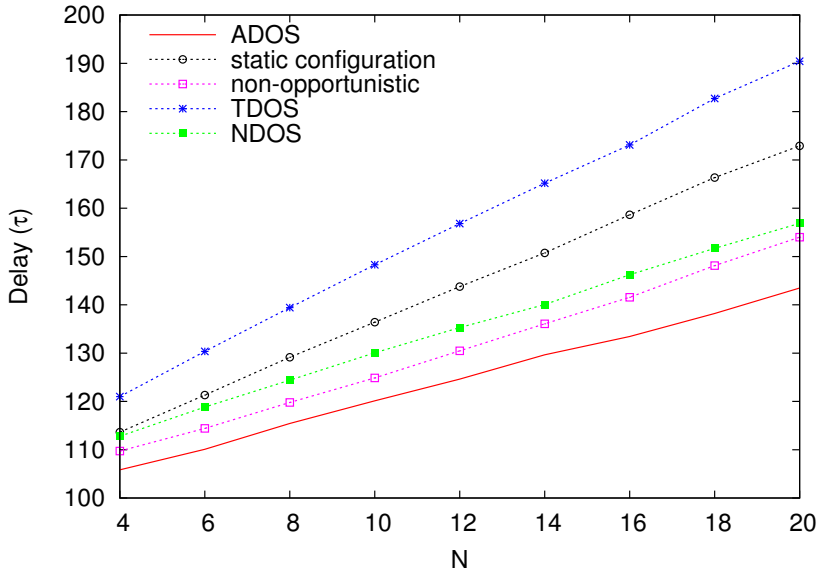


Figure 3.12: Delay performance.

In order to evaluate the delay performance of the proposed approach, Fig. 3.12 shows the average delay of the wireless network as a function of the number of stations for a homogeneous scenario ($\rho_i = 1\ \forall i$) when all stations transmit at half their saturation throughput. We observe from the figure that ADOS outperforms all the other approaches; it is particularly worth highlighting the ADOS even outperforms the '*non-opportunistic*' approach, which never gives up any transmission opportunity (but as a result, as transmissions are of fixed duration, requires performing more transmissions). We conclude from the results that ADOS does not only provide performance gains in terms of throughput but also in terms of delay.

3.6.6 Impact of channel coherence time

Our channel model is based on the assumption that different observations of the channel conditions are independent. In order to understand the impact of this assumption,

we repeated the experiment of Fig. 3.10a using *Jakes' channel model* [95] to obtain channel conditions that are correlated over time. The results, for a Doppler frequency of $f_D = 2\pi/100\tau$, are given in Fig. 3.13. We first observe that ADOS outperforms very substantially all the other approaches, which validates its effectiveness also in this case. We also observe that the throughput obtained is smaller than that of Fig. 3.10a. This is due to the fact that when the channel is bad, a station does not transmit after a successful contention, and therefore it takes a shorter time until it successfully contends again. As a result, a station accesses the channel more often when the channel is bad than when it is good, which introduces a bias that reduces throughput. We finally observe that performance lightly increases with the number of stations N, which is caused by the fact that the larger N, the less likely it is that a station that gave up a transmission opportunity wins a contention before its channel conditions improve.

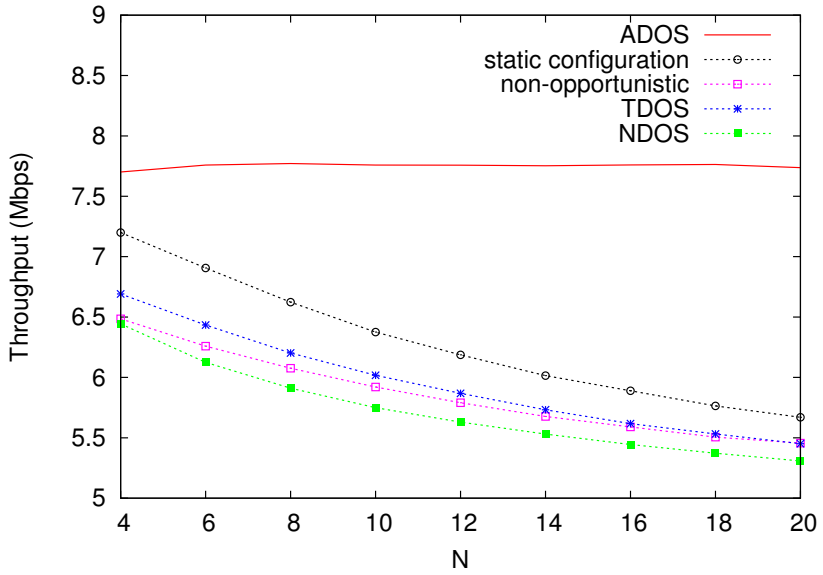


Figure 3.13: Performance with *Jakes' channel model*.

3.6.7 Discrete set of transmission rates

While all previous experiments assumed continuous rates, the design of ADOS do not rely on any assumption on the mapping of SNR to transmission rates, and therefore any mapping function (continuous or discrete) can be used. To show that ADOS is effective when only a set of discrete rates is allowed, we consider the case of a wireless system in which the only transmission rates available are $\{1, 2, 5.5, 12, 24, 48, 54\}$ Mbps. For a given SNR, we choose the largest available transmission rate that is smaller than the one given by (3.23). Fig. 3.14 shows the result of repeating the experiment of Fig. 3.10a with this discrete set of transmission rates. The results confirm that ADOS outperforms the other

approaches, and hence shows that the proposed mechanism also works well with discrete rates. Note that the resulting throughputs are lower, since for a given SNR, a station cannot use the maximum transmission rate supported by this SNR but needs to use a smaller one from the set of available rates.

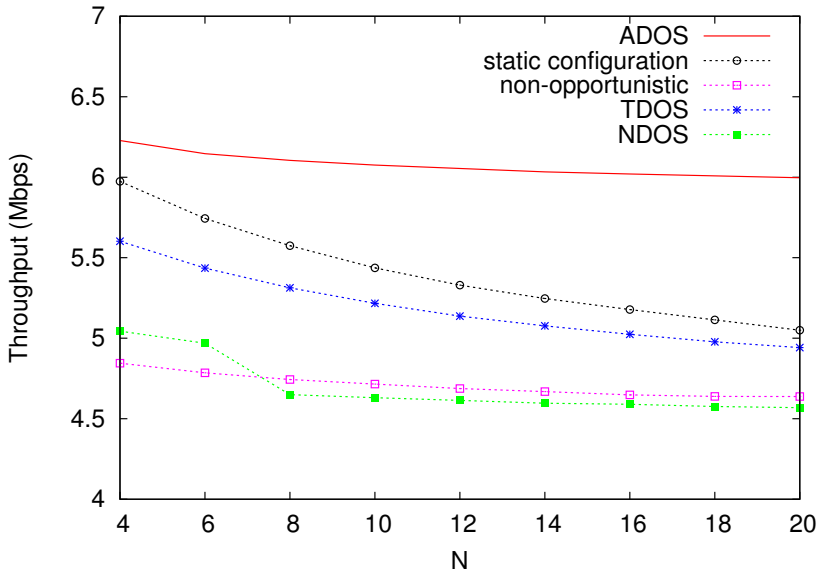


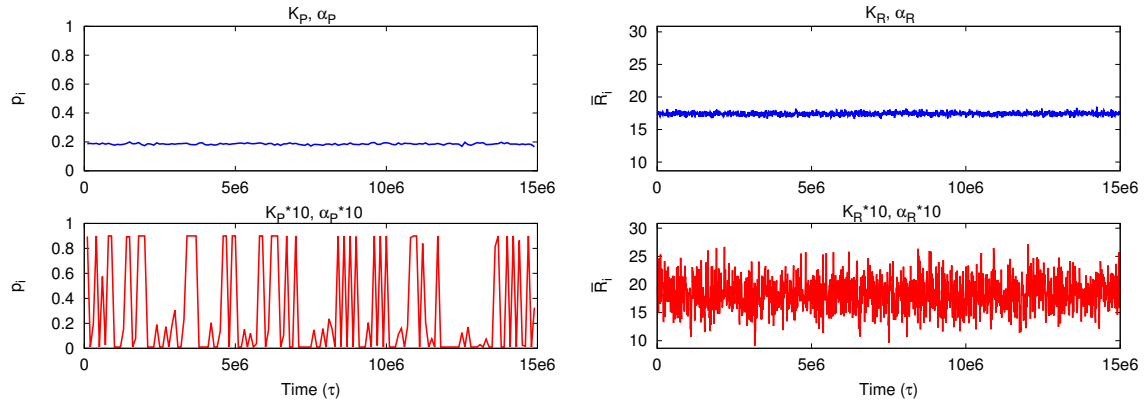
Figure 3.14: Throughput performance for a discrete set of rates.

3.6.8 Stability

The setting of the parameters $\{K_p, \alpha_p\}$ and $\{K_R, \alpha_R\}$ proposed in Section 3.5 achieves a good tradeoff between stability and speed of reaction. This is verified by the results presented in this and the following two sections.

To verify stable behavior, we first analyze the evolution over time of the access probability p_i of a station for the proposed $\{K_p, \alpha_p\}$ setting and for a configuration of these parameters 10 times larger, in a homogeneous scenario with $N = 5$ saturated stations and $\rho = 4$. Fig. 3.15a shows the evolution of p_i for both cases, sampled over $10^5\tau$ intervals. We observe from the figure that with the proposed setting (labeled "K_p, α_p"), p_i shows minor deviations around its average value, while for a larger setting (labeled "$K_p * 10, \alpha_p * 10$"), it shows unstable behavior with drastic oscillations.

Similarly, we also analyze the evolution over time of the threshold $\bar{R}_i$ of a station for the proposed $\{K_R, \alpha_R\}$ setting and for a configuration of these parameters 10 times larger in the same scenario. The results, depicted in Fig. 3.15b confirm that the proposed setting for these parameters is stable while a larger setting is highly unstable. We conclude from these results that the analysis conducted in Section 3.5 is effective in guaranteeing stability.



(a) Proposed $\{K_p, \alpha_p\}$ and for a larger setting.

(b) Proposed $\{K_R, \alpha_R\}$ and for a larger setting.

Figure 3.15: Stability evaluation

3.6.9 Changing number of stations

We next investigate the speed of reaction to changes in the number of stations of the network, which triggers the adjustment of the access probabilities p_i. To this aim, we consider a network with initially 5 stations, where 5 additional stations join the network after a time $5 \cdot 10^6 \tau$. All stations have $\rho = 4$. Fig. 3.16 shows the evolution of the access probability of one of the initial stations sampled over $10^5 \tau$ intervals. We observe from the figure that with our setting (labeled "K_p, α_p"), the system quickly adapts the p_i of the station to the new value. In contrast, for a setting of these parameters 10 times smaller (labeled "$K_p/10, \alpha_p/10$"), the reaction is very slow and it only converges after $5 \cdot 10^6 \tau$,

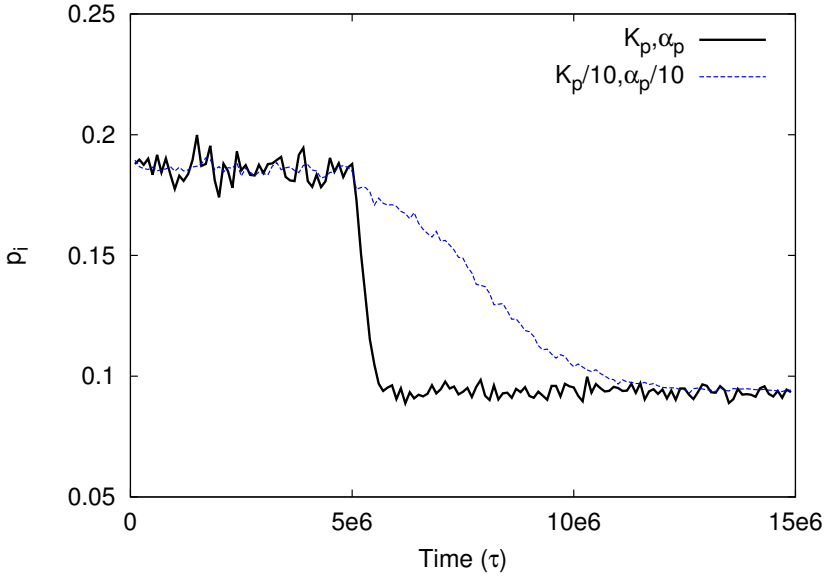


Figure 3.16: Speed of reaction: changing number of stations.

which clearly worsens the performance with respect to the configuration we proposed.

The results confirm that the proposed configuration for the parameters of the algorithm that adjusts p_i, $\{K_p, \alpha_p\}$, provides a good tradeoff between stability and speed of reaction, since with a larger setting of these parameters the system suffers from instability (as shown in the previous section), while with a smaller setting it reacts too slowly (as shown here).

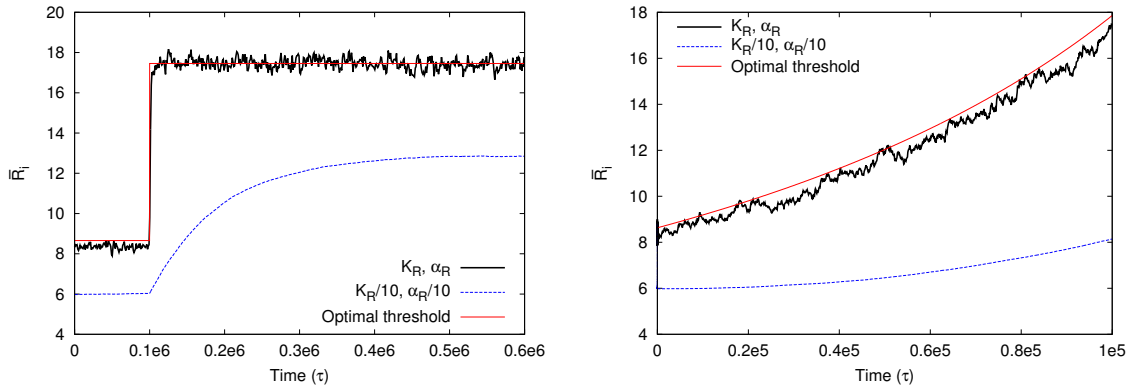
3.6.10 Changing radio conditions

To analyze the speed of reaction to changing radio conditions, we consider the following two scenarios: (i) a drastic change of the normalized SNR, which could be caused, e.g., by an obstacle, and (ii) a soft change of the normalized SNR caused, e.g., by the movement of the station. Both scenarios trigger the adjustment of $\bar{R}_i$; in the following, we study the evolution of $\bar{R}_i$ in each case.

For the first scenario, we consider that in a wireless network with $N = 2$ stations, both of them with a normalized SNR $\rho = 1$, one of the stations changes its normalized SNR to $\rho = 4$ after a time $5 \cdot 10^5 \tau$. Fig. 3.17a shows the evolution over time of the $\bar{R}_i$ of the station whose normalized SNR has changed, for the proposed setting of the $\{K_R, \alpha_R\}$ parameters as well as for a setting of these parameters 10 smaller. As a benchmark, the figure also shows the optimal setting of the threshold as given by the analytical results. The results show that: (i) with our setting of the parameters, the system reacts quickly and closely follows the benchmark, while the reaction is much slower for a smaller setting, and (ii) the steady state error with our setting is negligible, whereas with a smaller setting of the parameters it is much larger. The latter effect is caused by the fact that the steady error with a proportional controller increases as its proportional gain (K_R) is reduced. Therefore, by choosing a too small value for K_R, we do not only worsen the speed of reaction of the system but also its steady error.

For the second scenario, we consider that in a wireless with $N = 2$ stations, both of them with $\rho = 1$, one of the stations moves towards the sending station at a constant speed: initially, the station is located at a distance D (with an average normalized SNR of $\rho = 1$) and it moves to a distance $D/2$ of the sending station over a period of $10^5 \tau$. We consider a path loss exponent equal to 2. Fig. 3.17b shows the evolution of the $\bar{R}_i$ of the moving station over time. We observe that with our setting, the algorithm that adjusts $\bar{R}_i$ is able to cope with the movement of the station and $\bar{R}_i$ closely follows the optimal threshold. As in the previous case, with a smaller setting of the parameters the threshold used is far from the optimal because of the slow speed of reaction as well as the steady error.

Results confirm that the proposed configuration of the parameters $\{K_R, \alpha_R\}$ provides a good tradeoff between stability, speed of reaction and steady error: with a larger setting



(a) drastic change of ρ. (b) Speed of reaction: moving station.

Figure 3.17: Changing radio conditions

of the parameters the system suffers from instability, while with a smaller setting, it reacts too slowly and yields a large steady error.

3.6.11 Moving Stations

While the previous experiment involved only a single mobile station, in many cases some or all of the terminals may be moving. We next investigate a more complex scenario where $N = 10$ stations move in an area of size LxL following the random waypoint model, and send data to a station located at position (L,L). The transmission power is such that the normalized SNR for a station located at position (0,0) is $\rho = 1$.[7] We further consider a path loss exponent equal to 2. We compare ADOS against the following approaches: (i) a benchmark that uses, for the current normalized SNR, the optimal transmission rate threshold obtained from the analytical results ('*optimal*'), (ii) the '*non-opportunistic*' approach, (iii) TDOS, (iv) NDOS, and (v) the approach we proposed in [36] ('*static ADOS*').[8] For the TDOS, NDOS and *static ADOS* approaches, since they assume static radio conditions and hence rely on long term measurements to set the transmission rate threshold, we measure the average SNR over periods of $10^8\tau$ and use the measurement obtained in a period to compute the $\bar{R}_i$ of the next period.

Fig. 3.18 shows the performance of ADOS and the approaches described above in terms of the $\sum_i \log(r_i)$ averaged over intervals of $10^4\tau$, as a function of the speed with which stations move (in units of L/τ). By averaging the $\sum_i \log(r_i)$ over different time intervals, we not only capture the long-term fairness (i.e., fairness in total throughput)

[7]Note that we do not let ρ increase any further once a station is than a distance of $L/100$ to the receiver.

[8]Note that the approach proposed in this thesis differs from [36] in that it adapts to changing radio conditions; therefore, when radio conditions are static (as in experiments 3.6.1 to 3.6.9) both approaches behave in the same way.

but also the short-term fairness (i.e., fairness in the throughput obtained over a given time interval).[9] We observe from the results that the performance of ADOS closely follows the '*optimal*' benchmark and outperforms all other approaches. As in previous experiments, the '*non-opportunistic*', TDOS and NDOS approaches perform substantially worse that ADOS. The '*static ADOS*' approach also performs substantially worse, as it does not adjust to current radio conditions. While it does perform well for very low speeds for which the measurement period is sufficient, performance degrades sharply when the speed increases and stations far from the destination with an outdated threshold risk starvation. Performance improves slightly for even higher speeds, as the probability that a station stays far from the destination during the averaging period decreases, i.e., the threshold is outdated but due to the high speed the station is close the destination sufficiently often.

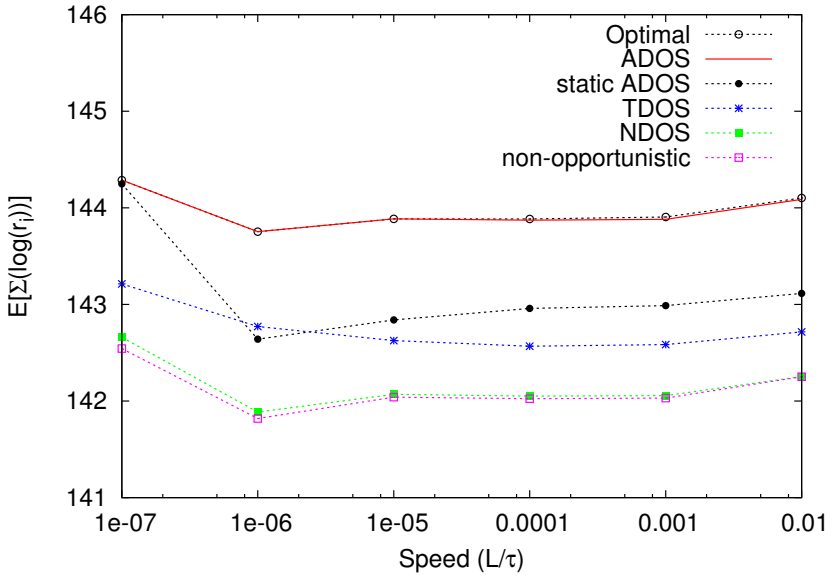


Figure 3.18: Moving stations.

3.7 Summary

The design of wireless communication systems over fading channels has typically been addressed by PHY layer mechanisms with the objective of increasing links' reliability. In contrast, more recent approaches have tackled this phenomenon by combining PHY/-MAC strategies to opportunistically schedule each user's transmissions with the goal of *exploiting*, rather than mitigating, fading. Distributed Opportunistic Scheduling (DOS) techniques provide throughput gains in wireless networks without requiring a centralized scheduler. One of the challenges of these techniques is the design of an adaptive algorithm

[9]Note that in the previous experiments where radio conditions were static, short-term fairness was not an issue.

that adjusts the DOS parameters to their optimal value. In this chapter we have proposed a novel algorithm, named ADOS, with the following advantages:

(*i*) it jointly optimizes both the access probabilities and the transmission thresholds;

(*ii*) it provides a good tradeoff between total throughput and fairness;

(*iii*) it guarantees convergence and stability.

A major finding when computing the configuration of the optimal threshold is that it is independent of the access probabilities, which decouples the computation of thresholds from that of the access probabilities.

The performance of ADOS has been extensively evaluated via simulations. Results confirm that ADOS provides significantly better performance than previous proposals; in particular, key results are that ADOS outperforms other approaches substantially with non-saturated stations as well as with changing radio conditions.

Chapter 4

Thwarting Selfish Behaviors in DOS

If no constraints are imposed on the wireless network and stations are allowed to configure their $\{\mathbf{p}\ \bar{\mathbf{R}}\}$ parameters to maximize their own benefit, the network will not naturally tend to the optimum configuration derived in the previous chapter. In the following, by combining game theory and control theory, we design a mechanism that removes the incentive to *misbehave*, i.e., to not follow the configuration that optimizes the overall performance in order to obtain individual gains.

4.1 Motivation

We model the wireless system as a static game in which each station can choose its configuration without suffering any penalty. The following theorem characterizes the Nash equilibria of this game.

Theorem 2. *In the absence of penalties, there is at least one station that plays $p_i = 1$ in any Nash equilibrium.*

Proof. The proof is by contradiction. Let us assume that there is a Nash equilibrium such that $p_j \neq 1\ \forall j$.

If we consider one player i and take the partial derivative of its throughput r_i, we obtain

$$\frac{\partial r_i}{\partial p_i} = \frac{\prod_{j \neq i}(1-p_j) l_i \hat{T}_{-i}}{\left(p_i \hat{T}_i + (1-p_i)\hat{T}_{-i}\right)^2} > 0 \tag{4.1}$$

where $\hat{T}_i$ is the average duration during which the channel is occupied when station i transmits and $\hat{T}_{-i}$ is the average duration of a transmission or an empty mini slot when station i does not transmit.

From the above, it can be seen that the throughput r_i is a strictly increasing function of p_i. It follows from this that $\{p_i, \bar{R}_i\}$, with $p_i \neq 1$, is not the best strategy for player i given the configuration of the other stations, since station i could obtain a higher throughput by increasing p_i to 1 and using the same $\bar{R}_i$. The configuration $\{p_i, \bar{R}_i\}$, with $p_i \neq 1$, is therefore not a Nash equilibrium, which contradicts our initial assumption. □

Any of the above Nash equilibria are highly undesirable. If station i is the only one that plays $p_i = 1$, then player i achieves non-zero throughput while all other players have zero throughput. Conversely, if some other station j also plays $p_j = 1$, the result is a network collapse with all players obtaining zero throughput.

We conclude from the above that, in the absence of punishments, selfish behavior will severely degrade the performance of the wireless system. In the following we propose DOC ("Distributed Opportunistic scheduling with distributed Control"), a distributed punishment mechanism that satisfies the following properties: (*i*) when all stations implement the algorithm, it leads to the optimal configuration computed in Chapter 3, and (*ii*) a selfish station cannot obtain any gain by deviating from the algorithm.

4.2 Rationale behind the algorithm

Before presenting the algorithm, we first discuss the rationale behind its design. This rationale relies heavily on the notion of *channel time* that a station obtains over a certain interval Θ, defined as

$$t_i(\Theta) = \sum_{k=1}^{n_i(\Theta)} \Big(T_i^k(\Theta) + (e-1)\tau \Big) \tag{4.2}$$

where $n_i(\Theta)$ is the number of successful contentions of station i in that period and $T_i^k(\Theta)$ is the duration of the k^{th} successful contention of the station in the interval. The above definition comprises the aggregate transmission time of the station plus a fixed overhead of $(e-1)\tau$ that is added every time the station accesses the channel.

An important observation that drives the design of our algorithm is that, with the configuration of Chapter 3, all stations receive the same channel time on average, i.e., $t_i = t_j \ \forall i, j$ (where $t_i = \mathbb{E}[t_i(\Theta)]$). This can be seen as follows. From (4.2) we have

$$\frac{t_i}{t_j} = \frac{\mathbb{E}[n_i(\Theta)] \left(\mathbb{E}[T_i^k(\Theta)] + (e-1)\tau \right)}{\mathbb{E}[n_j(\Theta)] \left(\mathbb{E}[T_j^k(\Theta)] + (e-1)\tau \right)} = \frac{p_{s,i}(T_i + (e-1)\tau)}{p_{s,j}(T_j + (e-1)\tau)}. \tag{4.3}$$

Furthermore, from (3.9) we have $p_{s,i}(T_i + (e-1)\tau) = p_{s,j}(T_j + (e-1)\tau)$ and thus $t_i = t_j$.

When all stations use the optimal configuration, the overhead in the definition of channel time, $(e-1)\tau$, coincides with the average time between two successes. As a

result, for an interval Θ of duration $T_{interval}$ it holds that $\sum_i \mathbb{E}[t_i(\Theta)] = T_{interval}$. From this and $t_i = t_j$, we have that with the optimal configuration, all stations receive an average channel time of

$$\mathbb{E}[t_i(\Theta)] = T_{interval}/N \qquad \forall i. \tag{4.4}$$

We define this average channel time as the *optimal channel time* and denote it by $t^* = T_{interval}/N$.

The last observation upon which our algorithm relies is that as long as a selfish station does not receive more channel time than t^*, it cannot increase its throughput. The throughput of a station with a given channel time and $\bar{R}_i$ is equal to the throughput it would obtain if it were alone in the channel during this time with $p_i = 1/e$ and the same $\bar{R}_i$. From Theorem 1, we have that this throughput is maximized for the optimal transmission rate threshold $\bar{R}_i^*$. Therefore, as long as the station does not receive extra channel time, it will not be able to achieve a higher throughput.

Given these observations, we base our algorithm on the following principles: (i) if a given station i detects that another station k is receiving more channel time than itself, it considers station k to be selfish and indirectly punishes it by using a more aggressive configuration, and (ii) when punishing station k, the punishment needs to be severe enough to keep station k's channel time below t^* so that station k does not benefit from misbehaving.

4.3 DOC algorithm

The first objective of DOC is to drive the system to the optimal configuration $\{\mathbf{p}^*, \bar{\mathbf{R}}^*\}$ obtained in Chapter 3, where each station can locally compute its optimal configuration of $\bar{R}_i$ independently of the configuration of the other stations. The second objective is to remove any incentive to misbehave discussed at the beginning of this chapter. To this aim, in DOC each station implements the same algorithm to compute the optimal $\bar{R}_i$ used in ADOS (see Chapter 3, and another algorithm to configure its access probability p_i. The latter is presented in the following.

With DOC, time is divided into intervals of fixed length $T_{interval}$, and each station updates its access probability p_i at the beginning of every interval. We use the discrete variable Θ to refer to the different intervals, and $p_i(\Theta)$ to denote the value of p_i in a given interval Θ. The central idea behind DOC is that when a misbehaving station is detected, the other stations increase their access probabilities in subsequent intervals to prevent the selfish station from benefiting from its misbehavior.

A key challenge in DOC is to determine the appropriate reaction against a selfish station. If the reaction is not severe enough, a selfish station may benefit from misbehaving.

However, if the reaction is too severe, the system may become unstable by entering an endless loop where all stations indefinitely increase their p_i to punish each other.

Like in the scenario considered in Chapter 3, control theory is a particularly suitable tool to address this challenge, as it helps guarantee the convergence and stability of adaptive algorithms. We use techniques from *multi-variable control theory* [105] for the design of the DOC algorithm. The algorithm is based on the classic system illustrated in Fig. 4.1, where each station runs an independent controller to compute its configuration. The controller that we have chosen for DOC is a *proportional-integral* (PI) controller, a well-known controller from classic control theory.

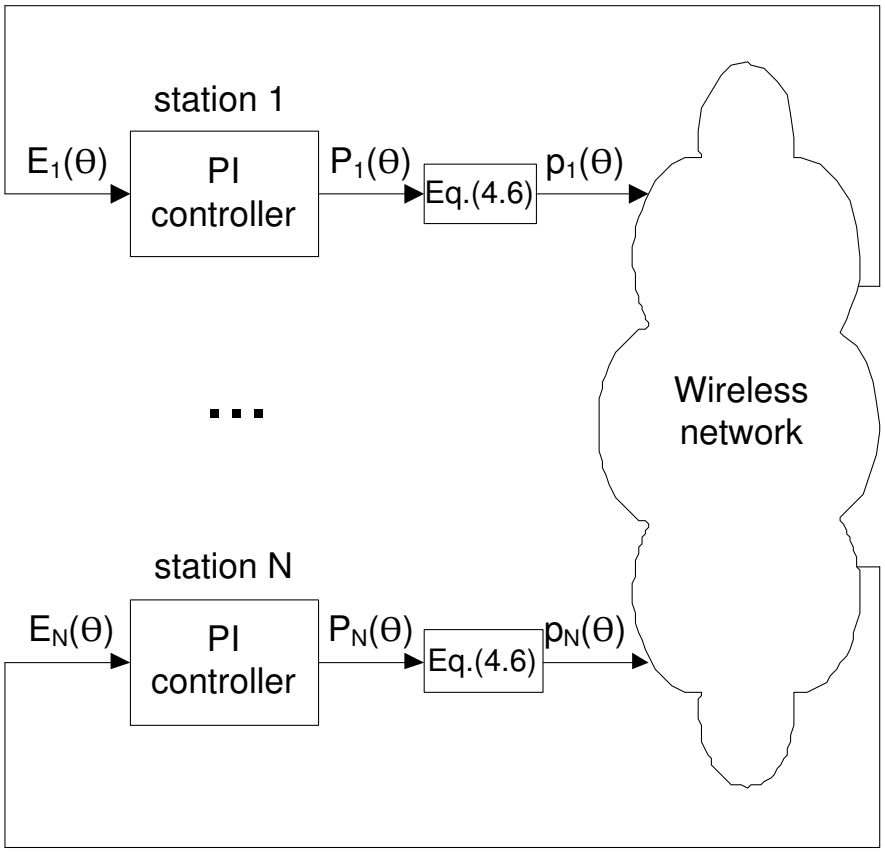


Figure 4.1: DOC control system.

As shown in the figure, the PI controller of station i takes as input the error signal measured over an interval Θ, $E_i(\Theta)$, and provides as output the control signal $P_i(\Theta)$ for the next interval. The error signal indicates how far the system is from the desired point of operation. If the system is operating as desired, the error signals of all stations are zero; otherwise, the error signals are non-zero and the state of the system needs to change from its current point of operation to the desired one. To do this, the PI controller adjusts the control signal $P_i(\Theta)$, increasing it if $E_i(\Theta) > 0$ and decreasing it otherwise. In the following, we address the design of $P_i(\Theta)$ and $E_i(\Theta)$.

4.3.1 Control signal P_i

The DOC algorithm needs to adjust the access probability $p_i(\Theta)$ based on the control signal. To do this, there needs to be a one-to-one mapping between the control signal $P_i(\Theta)$ given by the controller and $p_i(\Theta)$. In addition, we design the system such that the $P_i(\Theta)$ values are the same for all stations at the optimal point of operation. This latter requirement is necessary to later derive the conditions for stability in Section 4.4.

Based on the above requirements, we design $P_i(\Theta)$ as

$$P_i(\Theta) = \frac{p_i(\Theta)}{1 - p_i(\Theta)} \left(T_i + (e-1)\tau\right). \tag{4.5}$$

A station can therefore compute its $p_i(\Theta)$ from the control signal $P_i(\Theta)$ as

$$p_i(\Theta) = \frac{P_i(\Theta)}{T_i + (e-1)\tau + P_i(\Theta)}. \tag{4.6}$$

4.3.2 Error signal E_i

The design of the error signal $E_i(\Theta)$ has the following two goals: (i) selfish stations should not be able to obtain extra channel time from the wireless network by using a configuration different from the optimal, and (ii) as long as there are no selfish stations, $\mathbf{p}(\Theta)$ should converge to the optimal $\mathbf{p}^*$.

For the design of the error signal, DOC relies on the broadcast nature of the wireless medium, which enables stations to overhear the transmissions of the other stations (like [21, 22]). In particular, in every interval Θ, each station measures (i) the channel time used by the other stations, $t_j(\Theta)$, and (ii) the average time (over the interval) that they hold the channel upon a successful contention, $T_j(\Theta) = \sum_{k=1}^{n_j(\Theta)} T_j^k(\Theta)/n_j(\Theta)$. Based on this, station i computes the error signal at the end of the interval as

$$E_i(\Theta) = \sum_{j \neq i} \left(t_j(\Theta) - t_i(\Theta)\right) - F_i(\Theta) \tag{4.7}$$

where $F_i(\Theta)$ is a function that we design below. The error signal $E_i(\Theta)$ consists of the following two components:

- The first component, $\sum_{j \neq i} t_j(\Theta) - t_i(\Theta)$, punishes selfish stations. If a station i receives less channel time than the other stations, this component will be positive and hence station i will increase its access probability $p_i(\Theta)$.
- The second component, $F_i(\Theta)$, drives the system to the desired point of operation in the absence of selfish behavior (i.e., when all stations receive the same channel time).

We next address the design of the function $F_i(\Theta)$. In order to drive $\mathbf{p}(\Theta)$ to the desired $\mathbf{p}^*$ when all stations receive the same channel time, we need $F_i(\Theta) > 0$ for $p_i(\Theta) > p_i^*$, such that in this case $p_i(\Theta)$ decreases, and $F_i(\Theta) < 0$ for $p_i(\Theta) < p_i^*$.

The design of $F_i(\Theta)$ should also prevent selfish stations from obtaining more channel time than t^*. In the following, we derive the conditions that $F_i(\Theta)$ needs to meet in order to satisfy this requirement. To derive these conditions, we assume that the system is in steady state, which implies that selfish stations play with a static configuration. (In the analysis of Section 4.5 we show that DOC is also effective against selfish strategies that change the configuration over time.)

We first consider the case where one station k is selfish and all others are well-behaved and run the DOC algorithm. Since the PI controller drives the error signal $E_i(\Theta)$ to 0 in steady state, the following holds for all well-behaved stations:

$$F_i(\Theta) = \sum_{j \neq i} t_j(\Theta) - t_i(\Theta). \tag{4.8}$$

Summing $F_i(\Theta)$ over all stations except the selfish one yields:

$$\sum_{i \neq k} F_i(\Theta) = (N-1)t_k(\Theta) - \sum_{i \neq k} t_i(\Theta) = Nt_k(\Theta) - \sum_i t_i(\Theta). \tag{4.9}$$

If we combine the above with the requirement that the selfish station cannot gain, i.e., $t_k(\Theta) \leq t^*$, we obtain the following inequality,

$$\sum_{i \neq k} F_i(\Theta) \leq D(\Theta) \tag{4.10}$$

where $D(\Theta)$ is defined as the difference between the sum of channel times in optimal operation and the sum of channel times in the current interval, i.e., $D(\Theta) = Nt^* - \sum_i t_i(\Theta)$. Note that, if the current access probabilities are not optimal, $\sum_i t_i(\Theta)$ will be smaller than Nt^*. Hence, $D(\Theta)$ reflects the channel time lost due to non-optimal access probabilities.

The following upper bound on $F_i(\Theta)$ guarantees that (4.10) is satisfied, and thus ensures that a selfish station does not benefit from misbehaving:

$$F_i(\Theta) \leq \frac{1}{N-1} D(\Theta). \tag{4.11}$$

The intuition behind this upper bound is as follows. When a selfish station misbehaves, it receives more channel time than the well-behaved stations. This, however, moves the point of operation away from the optimal access probabilities, reducing the overall

efficiency in terms of aggregate channel time. The above upper bound ensures that the additional channel time received by the selfish station does not outweigh the channel time it loses due to the overall loss of aggregate channel time. This guarantees that the selfish station does not receive more channel time and hence does not benefit from misbehaving.

We next consider the case of multiple selfish stations. In this case, the aggregate channel time received by the selfish stations must not exceed the aggregate channel time that they would receive in optimal operation, i.e., $\sum_{i=1}^{m} t_i(\Theta) \leq mt^*$ (where $\{1, \ldots, m\}$ is the set of selfish stations). Following similar reasoning to that above, we obtain the upper bound

$$F_i(\Theta) \leq \frac{m}{N-m} D(\Theta). \tag{4.12}$$

Given all the above requirements, we design $F_i(\Theta)$ as:

$$F_i(\Theta) = \begin{cases} \min\left((N-1)D(\Theta), \frac{D(\Theta)}{N}\right), & p_i(\Theta) > p_i^{min} \\ \min\left((N-1)D(\Theta), -\frac{D(\Theta)}{N}, (N-1)\Delta\right), & p_i(\Theta) \leq p_i^{min} \end{cases} \tag{4.13}$$

where $\mathbf{p^{min}} = \{p_1^{min}, \ldots, p_N^{min}\}$ are the access probabilities that minimize $D = \mathbb{E}[D(\Theta)]$ subject to $t_i = t_j \ \forall i, j$, and Δ is the value that D takes at this point,

$$\Delta = D|_{\mathbf{p}=\mathbf{p^{min}}}. \tag{4.14}$$

In order to compute $\mathbf{p^{min}}$ and Δ, the T_j of all stations are required. For these, we use the $T_j(\Theta)$ values measured over the current interval.

The above design satisfies all of our previous requirements:

- The term $D(\Theta)/N$ ensures that (4.11) and (4.12) are satisfied when $D(\Theta) > 0$ and the term $(N-1)D(\Theta)$ ensures that they are satisfied when $D(\Theta) < 0$. This provides the required protection against (one or more) selfish stations.
- As illustrated in Fig. 4.2, when all stations have the same expected channel time, the expected value of $F_i(\Theta)$ is positive for $p_i(\Theta) > p_i^*$ and negative otherwise. This ensures that $\mathbf{p}(\Theta)$ is driven to the desired $\mathbf{p}^*$.

The above design of the DOC algorithm is based on the assumption that the number of stations in the wireless network is fixed. In the following, we address the case of stations joining and leaving the network. With DOC, each station only keeps the state maintained by the PI controller, $\sum_{\Theta} \sum_{j \neq i} \left(t_j(\Theta) - t_i(\Theta)\right) + F_i(\Theta)$, which accounts for the deficit or surplus of the station's channel time over the other stations in the network. When a new station joins the wireless network, this station does not have a surplus or deficit, and

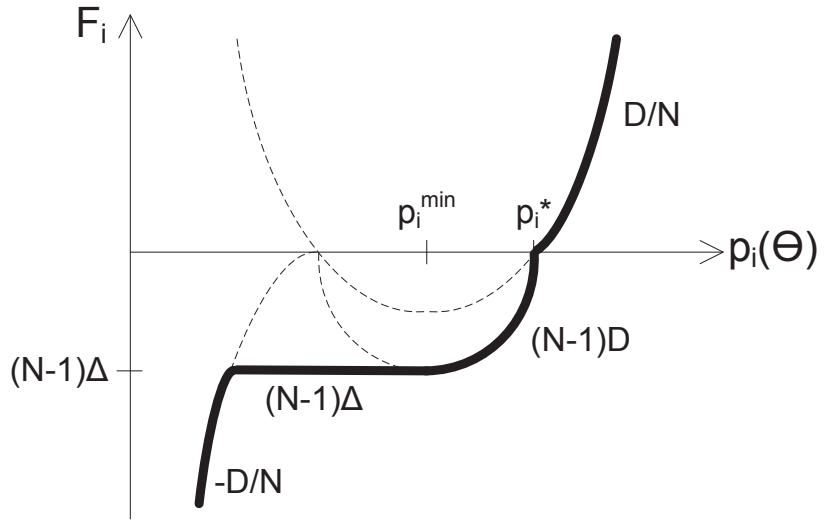


Figure 4.2: F_i as a function of $p_i(\Theta)$ when $t_i = t_j \ \forall i, j$.

therefore the other stations keep their state. The new station initializes the state of its PI controller such that its initial p_i corresponds to the optimal p_i^*. When a station leaves, the remaining stations keep their state: this ensures that the deficit accumulated by a selfish station is not reset if it leaves and rejoins the network.

This concludes the design of the algorithm. In the following two sections, we analytically evaluate its performance when all stations are well-behaved (Section 4.4) and when some stations misbehave (Section 4.5).

4.4 DOC analysis

In this section we analyze the performance of DOC when all stations are well-behaved. As stations do not obtain any benefit from misbehaving, it is to be expected that they will all play DOC, and therefore this is the most meaningful scenario for the performance analysis of the system. We first analyze the wireless system under steady state conditions and show that it is driven to the desired point of operation obtained in Sections 3.2 and 3.3. We then conduct a transient analysis and derive sufficient conditions for stability.

4.4.1 Steady state analysis

Our analysis is based on the system model of Fig. 4.3. In this model, C represents the function implemented by the controllers, which computes the control signals $P_i(\Theta)$, taking the error signals $E_i(\Theta)$ as input. H represents the wireless system which provides the error signals $E_i(\Theta)$ based on the control signals $P_i(\Theta)$. In line with standard control

theory [99], we model the randomness of the channel with the noise signals $W_i(\Theta)$, and let $E_i(\Theta)$ represent the expected value of the error signal for the given control signals $P_i(\Theta)$. Since the controller includes an integrator, there is no steady state error [99] and the steady state solution can be obtained from

$$E_i(\Theta) = 0 \qquad \forall i. \tag{4.15}$$

Using (4.7) and (4.13), $E_i(\Theta)$ can be computed from $\mathbf{p}(\Theta)$. This enables (4.15) to be expressed as a system of equations in $\mathbf{p}(\Theta)$. The following theorem guarantees that the the solution of this system of equations is unique and shows that the unique stable point in steady state is the desired point of operation from Sections 3.2 and 3.3.

Theorem 3. *The unique stable point of operation of the system in steady state is* $\mathbf{p}(\Theta) = \mathbf{p}^*$.

Proof. Let us consider two stations i and j. From (4.15) we have $E_i(\Theta) - E_j(\Theta) = 0$, which yields

$$Nt_j(\Theta) + F_j(\Theta) - Nt_i(\Theta) - F_i(\Theta) = 0. \tag{4.16}$$

Note that $t_j(\Theta) > t_i(\Theta)$ implies $F_j(\Theta) \geq F_i(\Theta)$, and vice versa. This can be seen as follows: If $p_j(\Theta) > p_j^{min}$ and $p_i(\Theta) > p_i^{min}$, then $F_j(\Theta) = F_i(\Theta)$. If $p_j(\Theta) \leq p_j^{min}$ and $p_i(\Theta) \leq p_i^{min}$, then also $F_j(\Theta) = F_i(\Theta)$. If $p_j(\Theta) > p_j^{min}$ and $p_i(\Theta) \leq p_i^{min}$, then $F_j(\Theta) \geq F_i(\Theta)$. When $t_j(\Theta) > t_i(\Theta)$, we are in one of these three cases, and hence $F_j(\Theta) \geq F_i(\Theta)$. Combining this with (4.16) yields $t_i(\Theta) = t_j(\Theta)\ \forall i, j$. Substituting this into $E_i(\Theta) = 0$ yields $F_i(\Theta) = 0$. Given $t_i(\Theta) = t_j(\Theta)$, $F_i(\Theta)$ is an increasing function of $p_i(\Theta)$ that crosses 0 at $p_i(\Theta) = p_i^*$. Hence, the only $p_i(\Theta)$ that satisfies $F_i(\Theta) = 0$ is p_i^*. Since this holds for all i, the unique stable point of operation is $p_i(\Theta) = p_i^*\ \forall i$. □

4.4.2 Stability analysis

We now conduct a stability analysis of DOC to configure the parameters of the PI controller. According to the definition of a PI controller [99], station i computes the value of P_i at interval Θ' as a function of the error values measured by the station in the current and previous intervals based on the following equation:

$$P_i(\Theta') = K_p E_i(\Theta') + K_i \sum_{\Theta=0}^{\Theta'-1} E_i(\Theta) \tag{4.17}$$

where K_p and K_i are the parameters of the controller that we need to configure.

In order to analyze our system from a control theoretic standpoint, we need to characterize the transfer functions C and H in the system model of Fig. 4.3. The control and

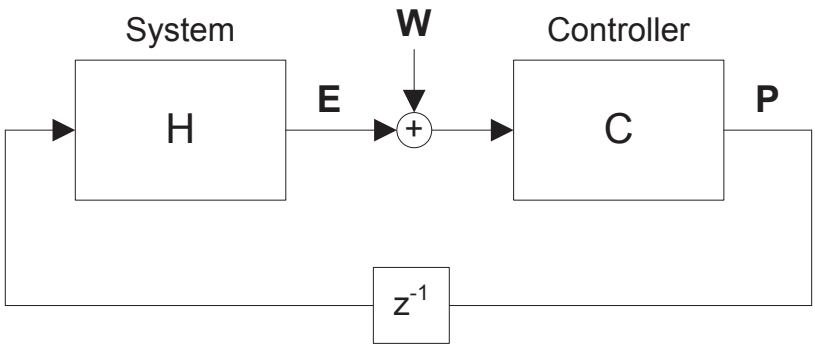


Figure 4.3: Control system.

error signals in the figure are given by the following vectors in the z-domain [99]:

$$\mathbf{P}(z) = (P_1(z), \ldots, P_N(z))^T \tag{4.18}$$

and

$$\mathbf{E}(z) = (E_1(z), \ldots, E_N(z))^T . \tag{4.19}$$

Our control system consists of one PI controller in each station i that takes $E_i(z)$ as input and provides $P_i(z)$ as output. We can therefore express the relationship between $\mathbf{E}(z)$ and $\mathbf{P}(z)$ as follows

$$\mathbf{P}(z) = C \cdot \mathbf{E}(z) \tag{4.20}$$

where

$$C = \begin{pmatrix} C_{PI}(z) & 0 & 0 & \ldots & 0 \\ 0 & C_{PI}(z) & 0 & \ldots & 0 \\ 0 & 0 & C_{PI}(z) & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & C_{PI}(z) \end{pmatrix} \tag{4.21}$$

with $C_{PI}(z)$ being the z-transform of a PI controller [99],

$$C_{PI}(z) = K_p + \frac{K_i}{z-1}. \tag{4.22}$$

In order to characterize our wireless system with a transfer function H that takes $\mathbf{P}(z)$ as input and has $\mathbf{E}(z)$ as output, we proceed as follows. Equation (4.7) provides a nonlinear relationship between $\mathbf{E}(\Theta)$ and $\mathbf{P}(\Theta)$. To express this relationship as a transfer function, we linearize it at the optimal point of operation.[1] We then study the linearized model and ensure its stability through appropriate choice of parameters. Note that the

[1]This linearization provides a good approximation of the behavior of the system when it suffers small perturbations around the stable point of operation.

stability of the linearized model guarantees that our system is locally stable (similarly as in Chapter 3).

We express the perturbations around the stable point of operation as follows:

$$\mathbf{P}(\Theta) = \mathbf{P}^* + \delta\mathbf{P}(\Theta) \tag{4.23}$$

where $\mathbf{P}^*$ is the stable point of operation as given by (4.5) with $\mathbf{p}(\Theta) = \mathbf{p}^*$.

With the above, the perturbations of $\mathbf{E}$ can be approximated by

$$\delta\mathbf{E}(\Theta) = H \cdot \delta\mathbf{P}(\Theta) \tag{4.24}$$

where

$$H = \begin{pmatrix} \frac{\partial E_1(\Theta)}{\partial P_1(\Theta)} & \frac{\partial E_1(\Theta)}{\partial P_2(\Theta)} & \cdots & \frac{\partial E_1(\Theta)}{\partial P_N(\Theta)} \\ \frac{\partial E_2(\Theta)}{\partial P_1(\Theta)} & \frac{\partial E_2(\Theta)}{\partial P_2(\Theta)} & \cdots & \frac{\partial E_2(\Theta)}{\partial P_N(\Theta)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial E_N(\Theta)}{\partial P_1(\Theta)} & \frac{\partial E_N(\Theta)}{\partial P_2(\Theta)} & \cdots & \frac{\partial E_N(\Theta)}{\partial P_N(\Theta)} \end{pmatrix}. \tag{4.25}$$

To compute these partial derivatives we proceed as follows. The error signal $E_i(\Theta)$ can be expressed as

$$\begin{aligned} E_i(\Theta) = T_{interval} \sum_{j \neq i} \Bigg(& \frac{p_{s,j}(\Theta)\,(T_j + (e-1)\,\tau)}{\sum_k p_{s,k}(\Theta) T_k + (1 - p_s(\Theta))\tau} \\ & - \frac{p_{s,i}(\Theta)\,(T_i + (e-1)\,\tau)}{\sum_k p_{s,k}(\Theta) T_k + (1 - p_s(\Theta))\tau} \Bigg) - F_i(\Theta). \end{aligned} \tag{4.26}$$

The above can be rewritten as a function of $\mathbf{P}(\Theta)$ given by

$$\begin{aligned} E_i(\Theta) = T_{interval} & \frac{\sum_{j \neq i} (P_j(\Theta) - P_i(\Theta))}{\sum_j P_j(\Theta) - \frac{p_s(\Theta)}{p_e(\Theta)}(e-1)\tau + \frac{1-p_s(\Theta)}{p_e(\Theta)}\tau} \\ & - F_i(\Theta) \end{aligned} \tag{4.27}$$

where $p_e(\Theta) = \prod_j 1 - p_j(\Theta)$.

We start by showing that $\partial F_i(\Theta)/\partial P_i(\Theta) = 0$ at the stable point of operation. It follows from (4.13) that

$$\frac{\partial F_i(\Theta)}{\partial P_i(\Theta)} = 0 \iff \frac{\partial D(\Theta)}{\partial P_i(\Theta)} = 0. \tag{4.28}$$

$D(\Theta)$ can be expressed as

$$D(\Theta) = Nt^* - T_{interval} \frac{\sum_i p_{s,i}(\Theta) T_i + p_s(\Theta)(e-1)\tau}{\sum_i p_{s,i}(\Theta) T_i + (1 - p_s(\Theta))\tau}. \tag{4.29}$$

The partial derivative of $D(\Theta)$ can be computed as

$$\frac{\partial D(\Theta)}{\partial P_i(\Theta)} = \frac{\partial D(\Theta)}{\partial p_i(\Theta)} \frac{\partial p_i(\Theta)}{\partial P_i(\Theta)}. \tag{4.30}$$

Taking the partial derivative of (4.29) with respect to $p_i(\Theta)$ and evaluating it at the stable point of operation yields

$$\frac{\partial D(\Theta)}{\partial p_i(\Theta)} = T_{interval} \left(\frac{\tau e}{\sum_i p_{s,i}(\Theta) T_i + (e-1)\tau} \right) \frac{\partial p_s(\Theta)}{\partial p_i(\Theta)}. \tag{4.31}$$

Since $p_s(\Theta)$ has a maximum value at the stable point of operation, we have that $\partial p_s(\Theta)/\partial p_i(\Theta) = 0$, which yields $\partial D(\Theta)/\partial P_i(\Theta) = 0$ and hence

$$\frac{\partial F_i(\Theta)}{\partial P_i(\Theta)} = 0. \tag{4.32}$$

The partial derivative of $E_i(\Theta)$ evaluated at the stable point of operation can then be computed from (4.27) as

$$\left. \frac{\partial E_i(\Theta)}{\partial P_i(\Theta)} \right|_{\mathbf{P}(\Theta)=\mathbf{P}^*} = -(N-1) T_{interval} \frac{1}{\sum_j P_j^*}. \tag{4.33}$$

Using similar reasoning, we can see that

$$\left. \frac{\partial E_j(\Theta)}{\partial P_j(\Theta)} \right|_{\mathbf{P}(\Theta)=\mathbf{P}^*} = T_{interval} \frac{1}{\sum_j P_j^*}. \tag{4.34}$$

Substituting these expressions in matrix H yields

$$H = K_H \begin{pmatrix} -(N-1) & 1 & \dots & 1 \\ 1 & -(N-1) & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & -(N-1) \end{pmatrix} \tag{4.35}$$

where

$$K_H = T_{interval} \frac{1}{\sum_j P_j^*}. \tag{4.36}$$

Thus, the linearized system is fully characterized by the matrices C and H. The next step is to configure the K_p and K_i parameters. The following theorem provides sufficient conditions which $\{K_p, K_i\}$ must meet to ensure stability:

Theorem 4. *The linearized system is guaranteed to be stable as long as K_p and K_i meet*

the following conditions:

$$K_i < K_p + \frac{1}{NK_H}, \quad K_i > 2K_p - \frac{1}{NK_H}. \tag{4.37}$$

Proof. The reader is referred to [106] for the proof of the theorem. Since [106] uses the same linearized system as DOC, the proof follows very closely that of [106]. □

In addition to guaranteeing stability, our goal in the configuration of the $\{K_p, K_i\}$ parameters is to find the right balance between reaction time in transients and oscillations in steady state. To this end, similarly as in Chapter 3, we use the *Ziegler-Nichols* rules [99], which have been designed for this purpose. Following these rules (see [106] for a detailed description), we obtain the configuration:

$$K_p = \frac{0.4}{2NK_H}, \quad K_i = \left(\frac{1}{0.85 \cdot 2}\right) \frac{0.4}{2NK_H}. \tag{4.38}$$

The stability of the resulting configuration is guaranteed by the following corollary:

Corollary 1. *The K_p and K_i configuration given by (4.38) is stable.*

Proof. The proof follows from the fact that the configuration of (4.38) meets the conditions of Theorem 4. □

Note that the above control theoretic analysis guarantees that the system will always converge to the desired point of operation regardless of the initial state. This implies that the system remains stable in the presence of any kind of perturbation. Such perturbations include, among others, transient selfish behavior or stations joining and leaving the network.

4.5 Game theoretic analysis

In the previous section we have shown that, when all stations implement the DOC algorithm, they all use $p_i = p_i^*$ and $\bar{R}_i = \bar{R}_i^*$, which leads to the optimal throughput allocation r_i^* obtained in Sections 3.2 and 3.3.[2] In this section we conduct a game theoretic analysis to show that one or more stations cannot obtain any gain by deviating from DOC. In what follows, we say that a station is *honest* (or well-behaved) when it implements the DOC algorithm to configure its p_i and $\bar{R}_i$ parameters, while we say that it is *selfish* or

[2]Since the throughput allocation $\{r_1^*, \ldots, r_N^*\}$ maximizes $\sum_i \log(r_i)$, it is *Pareto optimal*. This follows from the fact that if there existed another feasible allocation that provided all stations with more throughput than r_i^*, this allocation would yield a larger $\sum_i \log(r_i)$.

misbehaving when it plays a strategy different from DOC to configure these parameters in order to obtain a greater share of wireless resources.

The game theoretic analysis conducted in this section assumes that users are *rational* and want to maximize their own benefit, which is given by the throughput. Furthermore, it is reasonable to assume that the game is non-cooperative in that no binding agreements can be reached between the players as to their their future play [22]. The model is based on the theory of *repeated games* [74]. In repeated games, time is divided into stages and a player can take new decisions at each stage based on the observed behavior of the other players in the previous stages. This matches our algorithm, where time is divided into intervals and stations update their configuration at each interval.[3] Like previous analyses on repeated games [21, 22], we consider an infinitely repeated game, which is a common assumption when the players do not know when the game will end.

4.5.1 Single selfish station

While the design of the DOC algorithm in Section 4.4 guarantees that a station cannot benefit from playing with a *fixed* selfish configuration, selfish stations might still benefit by *varying* their configuration over time. As an example, let us consider a naïve algorithm that only takes into account the stations' behavior in the previous stage. While this algorithm may be effective against a fixed selfish configuration, it could easily be defeated by a selfish station that alternates between a selfish configuration ($p_k = 1, \bar{R}_k = 0$) and an honest one ($p_k = p_k^*, \bar{R}_k = \bar{R}_k^*$) at every other stage. Since this station would play selfish when all the others play honest, it would achieve a significantly higher throughput every other interval, thus benefiting from its misbehavior.

The above example shows that it is important to ensure that a selfish station cannot obtain any gain no matter how it varies its configuration over time. The following theorem confirms the effectiveness of DOC against any (fixed or variable) selfish strategy. The proof of the theorem relies on the integrator component of the PI controller, which keeps track of the aggregate channel time received by all stations and can thus be used to guarantee that this aggregate does not exceed a given amount.

Theorem 5. *Let us consider a selfish station that uses a $p_k(\Theta)$ and $\bar{R}_k(\Theta)$ configuration that can vary over time. If all the other stations implement the DOC algorithm, the throughput received by this station will be no larger than r_k^*.*

[3] Note that the game theoretic study conducted at the beginning of this chapter was based on static games instead of repeated ones. The reason is that we considered a system without penalties where a user does not react to the behavior of other users. Hence, we could model it as a static game where all players only make a single move at the beginning of the game.

Proof. The PI controller computes P_i at a given interval Θ' according to the following expression:

$$P_i(\Theta') = P_i^{initial} + K_p\Big(\sum_{j\neq i}(t_j(\Theta') - t_i(\Theta')) - F_i(\Theta')\Big) + K_i\sum_{\Theta=0}^{\Theta'-1}\Big(\sum_{j\neq i}(t_j(\Theta) - t_i(\Theta)) - F_i(\Theta)\Big). \tag{4.39}$$

With the above, $P_i(\Theta')$ stays between 0 and a given maximum value P_i^{max}. If at some point P_i reaches a P_i^{max} value such that $p_i = 1$, this will result in $t_j = 0$ for $j \neq i$ and $F_i > -(N-1)t_i$, which yields $E_i < 0$, and therefore P_i will decrease. Similarly, if at any time P_i reaches 0, then $t_i = 0$ and $F_i \leq 0$, which yields $E_i > 0$, and therefore P_i will increase.

Considering that $0 \leq P_i(\Theta') \leq P_i^{max}$, the above equation can be expressed as

$$\sum_{\Theta=0}^{\Theta'}\Big(\sum_{j\neq i}(t_j(\Theta) - t_i(\Theta)) - F_i(\Theta)\Big) = B_i \tag{4.40}$$

where B_i is a bounded value: $B_i = (P_i^{max} - P_i^{initial} + (K_i - K_p)E_i(\Theta'))/K_i$.

Let us consider the case in which there is a selfish station that changes its configuration over time and receives a channel time $t_k(\Theta)$. Equation (4.40) can be written as

$$\sum_{\Theta} t_k(\Theta) = \sum_{\Theta}\Big((N-1)t_i(\Theta) - \sum_{j\neq i,k} t_j(\Theta) + F_i(\Theta)\Big) + B_i. \tag{4.41}$$

Let us now consider a given interval Θ. From (4.11), we have

$$F_j(\Theta) \leq \frac{1}{N-1}\Big(Nt^* - \sum_i t_i(\Theta)\Big). \tag{4.42}$$

Summing the above expression over all $j \neq k$, we have $\sum_i t_i(\Theta) + \sum_{j\neq k} F_j(\Theta) \leq Nt^*$. As this satisfied for all Θ,

$$\sum_{\Theta}\Big(\sum_i t_i(\Theta) + \sum_{j\neq k} F_j(\Theta)\Big) \leq \sum_{\Theta} Nt^*. \tag{4.43}$$

Furthermore, by summing (4.41) over all $j \neq k$,

$$(N-1)\sum_{\Theta} t_k(\Theta) = \sum_{j\neq k}\sum_{\Theta}(t_j(\Theta) + F_j(\Theta)) + \sum_{j\neq k} B_j. \tag{4.44}$$

Adding the above two equations yields $N\sum_{\Theta} t_k(\Theta) \leq N\sum_{\Theta} t^* + \sum_{j\neq k} B_j$. If we consider a long period of time, the constant term $\sum_{j\neq k} B_j$ can be neglected, resulting in $\sum_{\Theta} t_k(\Theta) \leq \sum_{\Theta} t^*$. This means that the selfish station cannot receive more channel time using a selfish strategy than by playing DOC. Following the argument of Section 4.2, this implies that it cannot obtain more throughput than it would by playing DOC, i.e., $r_k \leq r_k^*$, which proves the theorem. □

The above theorem leads to Corollary 2.

Corollary 2. *A state in which all stations play DOC (*All-DOC*) is a Nash equilibrium of the game.*

Proof. According to Theorem 5, if all stations but one play DOC, the best response of this station is to play DOC as well since it cannot benefit from playing a different strategy. Thus, *All-DOC* is a Nash equilibrium. □

The above shows that if all stations start playing with no previous history, none of them can benefit by deviating from DOC. In addition to this, in repeated games it is also important to ensure that, if at some point the game has a given history, a selfish station cannot exploit knowledge of this history by playing a different strategy from DOC. The following theorem confirms that *All-DOC* is a Nash equilibrium of any subgame (where a subgame is defined as the game resulting from starting to play with a certain history). Therefore, a selfish station cannot benefit by deviating from DOC for any previous history of the game.

Theorem 6. *All-DOC is a subgame perfect Nash equilibrium of the game.*

Proof. Since the proof of Theorem 5 is not dependent on past history and can therefore be applied to any subgame, *All-DOC* is a Nash equilibrium of any subgame. □

Note that, even though the above analysis assumes a fixed number of stations in the wireless network, it also holds for the case when the number of stations changes over time, as long as these changes occur over sufficiently long periods such that the constant term $\sum_{j\neq k} B_j$ is not significant.

4.5.2 Multiple selfish stations

The above results show the effectiveness of DOC against a single selfish station. In the following, we focus on the case of multiple selfish stations.

The following theorem shows that, by following a different strategy from DOC, multiple stations cannot gain any aggregate channel time.

Theorem 7. *Let us consider a scenario with m selfish stations. If all other stations play DOC, the selfish stations cannot gain any aggregate channel time.*

Proof. Without loss of generality, let us consider that stations $i = \{1, \ldots, m\}$ are selfish. Applying a reasoning similar to Theorem 5 leads to $\sum_{i=1}^{m} \sum_{\Theta} t_i(\Theta) \leq m \sum_{\Theta} t^*$. As the left-hand side of this inequality is the aggregate channel time obtained by the selfish stations, and the right-hand side is the aggregate channel time that they would obtain if they played DOC, the theorem is proven. □

According to the above theorem, it is possible for a selfish station to obtain some gain, but this will be at the expense of another selfish station that receives less channel time. Corollary 3 follows from this.

Corollary 3. *Let us consider a scenario with m selfish stations. If all other stations play DOC and a selfish station k receives a throughput larger than r_k^*, then there exists another selfish station l that receives a throughput smaller than r_l^*.*

Proof. If there is some station $k \in \{1, \ldots, m\}$ for which $r_k > r_k^*$, this station must necessarily receive more channel time than it would if all stations played DOC. Since (according to Theorem 7) the selfish stations cannot gain any aggregate channel time, there must then exist some other station $l \in \{1, \ldots, m\}$ that receives less channel time. For this station, it holds that $r_l < r_l^*$, which proves the corollary. □

Based on the above, we argue that DOC is effective against multiple selfish stations, since two or more selfish stations cannot *simultaneously* benefit and therefore do not have any incentive to play a coordinated strategy different from DOC.

4.6 Performance evaluation

In this section we evaluate DOC by means of simulation to show that (*i*) in the absence of selfish stations, DOC provides optimal performance while remaining stable and reacting quickly to changes, and (*ii*) selfish stations cannot benefit by following a strategy different from DOC. We implement DOC in the same simulator used in Chapter 3, i.e., we assume that the available transmission rate for a given SNR is given by the Shannon channel capacity: $R(h) = W \log_2(1 + \rho|h|^2)$ bits/s, where W is the channel bandwidth, ρ is the normalized average SNR and h is the random gain of Rayleigh fading. In the simulations, we set $W = 10^7$, $\mathcal{T}/\tau = 10$ and the interval of the controller $T_{total} = 10^5\tau$. As in the prior chapter, for all results, 95% confidence intervals are below 1%.

4.6.1 Throughput evaluation

For the throughput evaluation, we compare the performance of DOC to the following approaches: (*i*) the static optimal configuration obtained in Chapter 3 ('*optimal configuration*'), (*ii*) the team game approach proposed in [7][4] ('*TDOS*'), and (*iii*) an approach that does not perform opportunistic scheduling but always transmits after successful contention ('*non-opportunistic*')[5]. We consider a scenario with $N = 10$ stations, half of them with a normalized SNR of $\rho_1 = 1$ and the other half with a normalized SNR ρ_2 that varies from 1 to 10. Fig. 4.4 shows the proportional fairness metric, $\sum_i \log(r_i)$, as a function of ρ_2. We observe that DOC performs at the same level as the benchmark given by the *optimal configuration*, while the other two approaches (*TDOS* and *non-opportunistic*) provide a substantially lower performance.

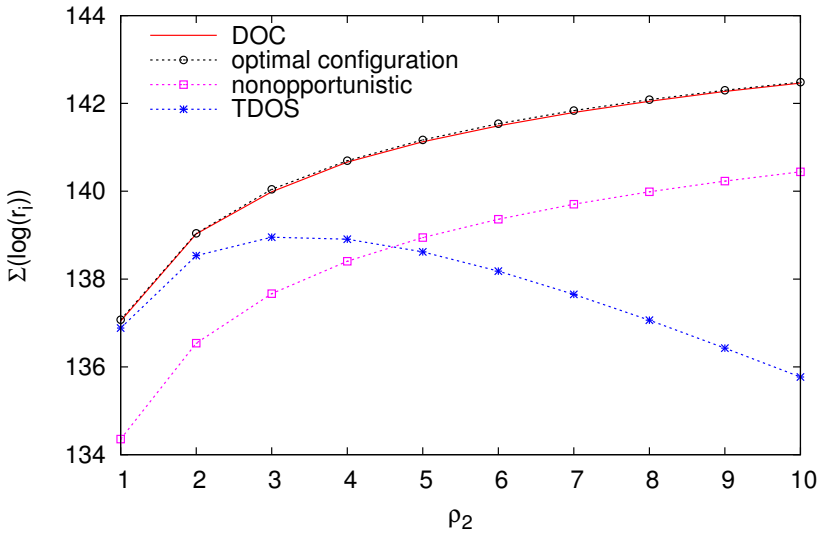


Figure 4.4: Proportional fairness as a function of SNR ($\rho_1 = 1, 1 \leq \rho_2 \leq 10$).

For the above scenario with $\rho_2 = 4$, Fig. 4.5 depicts the individual throughput allocation of two stations (where r_1 is the throughput of a station with ρ_1, and r_2 that of a station with ρ_2). DOC is effective in driving the system to the optimal point of operation and provides the same throughput as the *optimal configuration*. In contrast, *TDOS* exhibits a high degree of unfairness as it provides a much higher throughput to the station with high SNR. The *non-opportunistic* approach provides a reasonable degree of fairness but has lower throughput due to the lack of opportunistic scheduling. In conclusion, the proposed DOC algorithm provides a good tradeoff between overall throughput and

[4]Note that this approach requires that each station knows the channel state of all stations in the network.

[5]Note that we do not compare with the non-cooperative approach ('NDOS') proposed in [7] throughout this chapter. The reason is that, as shown in Chapter 3, the performance is well below the optimal configuration achieved by DOC and therefore we do not include it to simplify the benchmarking.

fairness.

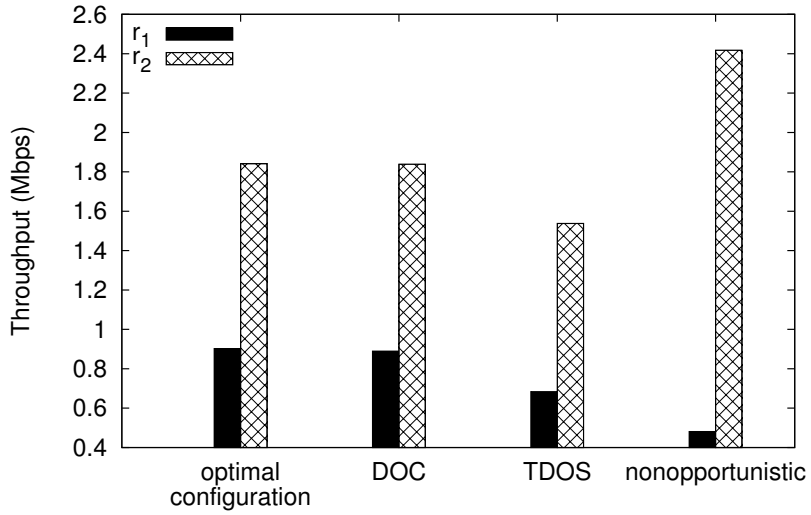


Figure 4.5: Throughput for heterogeneous SNRs ($\rho_1 = 1, \rho_2 = 4$).

4.6.2 Selfish station with fixed configuration

We verify that a station cannot obtain more throughput with a selfish configuration than by playing DOC in a scenario with $N = 10$ stations, half of them with $\rho_1 = 1$ and the other half (including the selfish station) with $\rho_2 = 4$. The selfish station uses a fixed configuration and all other stations implement DOC. Fig. 4.6 shows the throughput of the selfish station for different configurations $\{p_k, \bar{R}_k\}$ of the selfish station. This is compared to the throughput that it would obtain if it played DOC, given by the horizontal line. We observe that none of the selfish configurations provides greater throughput than DOC.

Fig. 4.7 analyzes the impact of fixed selfish configurations for a range of different N and ρ_2 values. It shows the largest throughput that a selfish station can receive with a fixed configuration, which is obtained by performing an exhaustive search over the $\{p_k, \bar{R}_k\}$ space. This throughput is compared to that which station would receive if it played DOC. Again, we observe that the station never benefits from playing selfishly, which validates the design of the DOC algorithm.

4.6.3 Selfish station with variable configuration

According to Theorem 5, a selfish station cannot benefit from changing its configuration over time. To verify this, we evaluate the throughput obtained by a selfish station with different adaptive strategies. These strategies are inspired by the schemes used in [21]

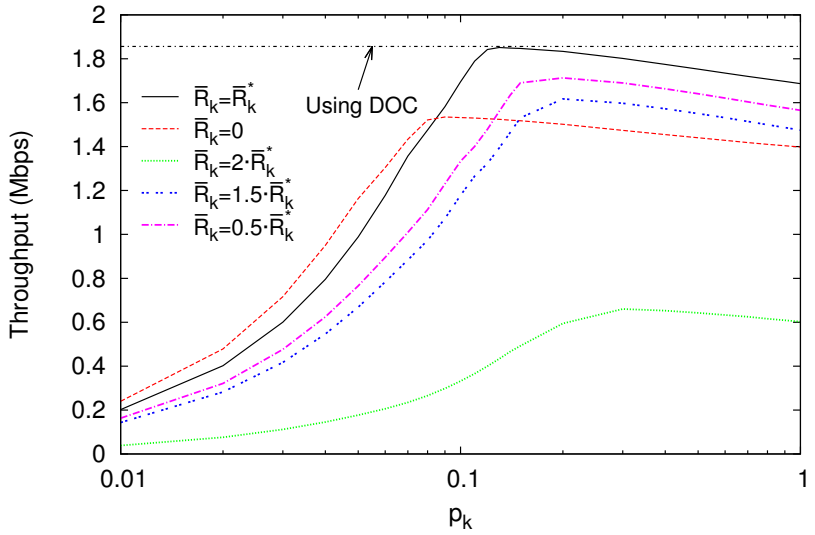


Figure 4.6: Throughput of a selfish station for fixed configurations of $\{p_k, \bar{R}_k\}$.

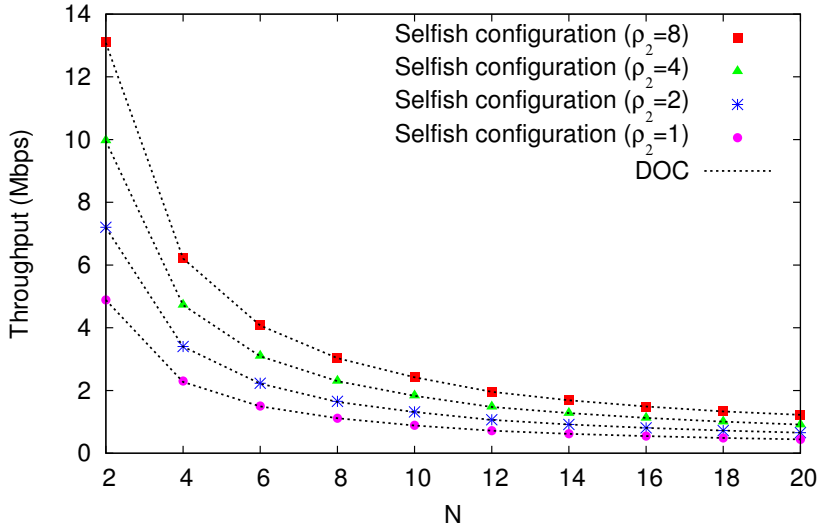


Figure 4.7: Selfish station with fixed configuration for different N and ρ_2 values.

for a similar purpose. The underlying principle of all of them is that the cheating station uses a selfish configuration to gain more throughput and, when it realizes that it is not obtaining more throughput, it assumes that it has been detected as selfish and switches back to the honest configuration to avoid being punished.

In particular, we consider the following strategies:

- The '*adaptive p_k strategy*'. This strategy fixes the $\bar{R}_k$ configuration of the selfish station to its optimal value, $\bar{R}_k = \bar{R}_k^*$, and modifies the p_k configuration as follows: the station uses a selfish configuration of $p_k = 1$ as long as it obtains some gain,

i.e. $r_k > r_k^*$. When r_k drops below r_k^*, the station switches to the honest configuration, $p_k = p_k^*$, and stays with this configuration as long as r_k remains below $0.95r_k^*$. It switches back to $p_k = 1$ when r_k exceeds $0.95r_k^*$.

- The '*adaptive* $\bar{R}_k$ *strategy*'. This strategy fixes the p_k configuration to the optimal value, $p_k = p_k^*$, and modifies the $\bar{R}_k$ configuration following a strategy similar to the one above: the station uses a selfish configuration of $\bar{R}_k = 0$ (i.e., it uses all transmission opportunities) as long as it obtains some gain and switches to the honest configuration when it stops benefiting.

- The '*adaptive* p_k *and* $\bar{R}_k$ *strategy*'. This strategy follows a similar behavior to the previous ones but adapts the configuration of both p_k and $\bar{R}_k$.

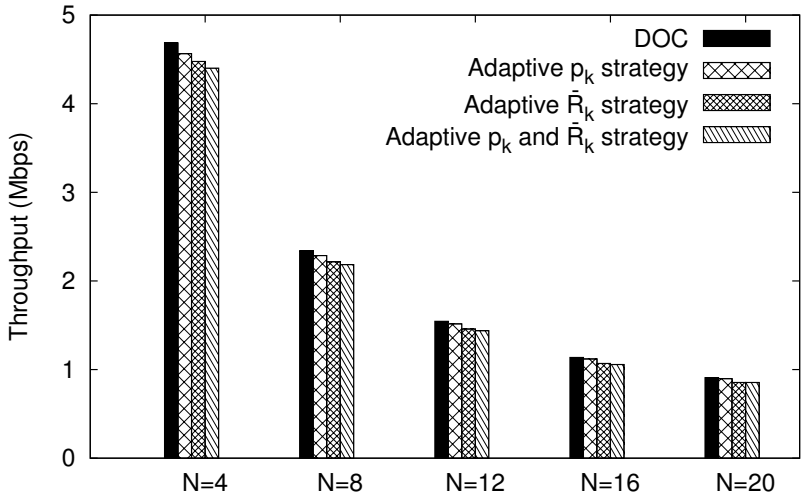


Figure 4.8: Throughput of selfish station with different adaptive strategies.

Fig. 4.8 compares the throughput obtained with each of the above strategies to that obtained with DOC for different values of N. As expected, when all other stations play DOC, a given station maximizes its payoff by playing DOC as well, as this results in a larger throughput for the station than any of the other strategies. This confirms the result of Theorem 5.

4.6.4 Multiple selfish stations

Corollary 3 states that all of the selfish stations cannot simultaneously benefit by deviating from DOC: if one or more of the selfish stations experience throughput gains, there must be other selfish stations that suffer some loss.

To validate the result, we consider a network with $N = 10$ stations (two of them selfish). Half of the stations (including one of the selfish stations) have $\rho_1 = 1$ and the other half (including the other selfish station) have $\rho_2 = 4$. We perform an exhaustive search over a wide range of $\{p_i, \bar{R}_i\}$ configurations of the two selfish stations. The results of this experiment are depicted in Fig. 4.9, which shows the throughput obtained by the two selfish stations (r_k and r_l) for each of the configurations used in the exhaustive search. The figure also shows the throughput of the two stations when they both play DOC. There is no configuration that simultaneously improves the throughput of the two selfish stations, which confirms the result of Corollary 3.

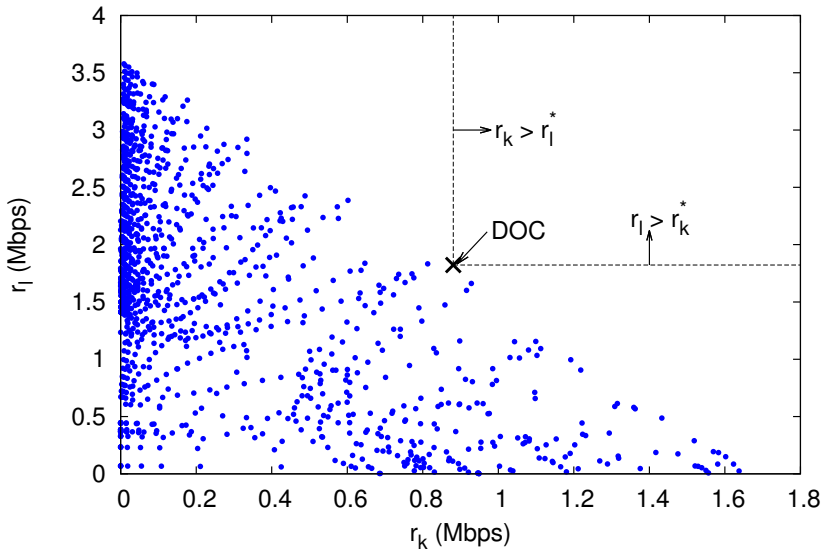


Figure 4.9: Throughput obtained by multiple selfish stations.

We also observe from the figure that the region of feasible allocations has a triangular shape. This is a consequence of Theorem 7: since the maximum aggregate channel time that the two stations can obtain is fixed, any throughput increase in one station leads to a decrease in the other station of the same amount, scaled by a constant factor that depends on the respective radio conditions.

4.6.5 Setting of the parameters of the PI controller

The main objective in the configuration of the K_p and K_i parameters proposed in Section 4.4 is to achieve a good tradeoff between stability and reaction time. To validate that our system guarantees stable behavior, we analyze the evolution of the throughput received by a station over time in a wireless network with $N = 10$ stations. Fig. 4.10 shows the throughput for the chosen setting (labeled "K_p, K_i") and for a configuration of these parameters 10 times larger (labeled "$K_p * 10, K_i * 10$"). We observe that with the

proposed setting, the throughput only suffers minor deviations around its average value. In contrast, for a larger setting, it exhibits highly oscillatory, unstable behavior.

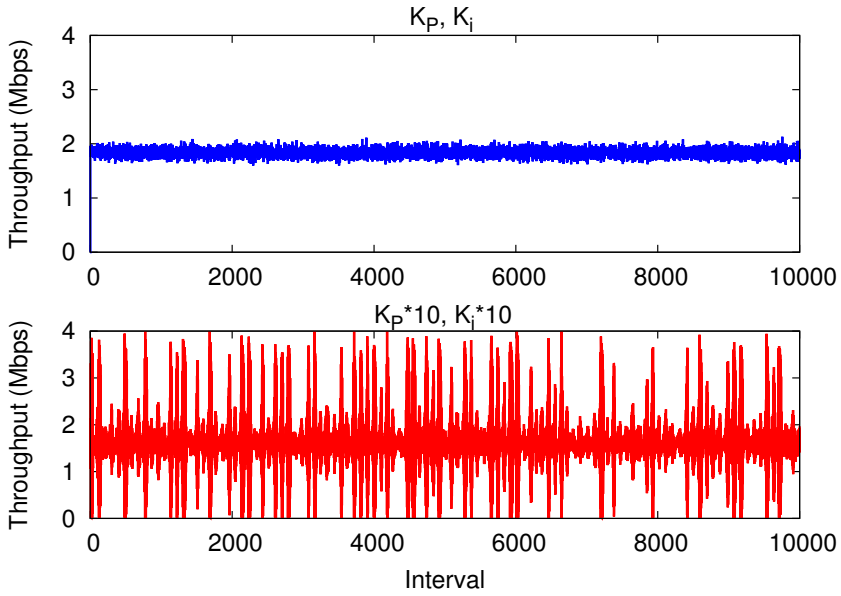


Figure 4.10: Stability analysis of the parameters of the PI controller.

To investigate DOC's reaction speed, we consider the following scenario. In a wireless network with $N = 10$ stations, initially all stations play DOC. After 50 intervals, one station becomes selfish and changes its access probability to $p_k = 1$. Fig. 4.11 shows the evolution of the throughput of the selfish station over time. We observe from the figure that with our setting (labeled "K_p, K_i"), the system reacts quickly, and after a few tens of intervals the selfish station no longer benefits from its behavior. In contrast, for a parameter setting 10 times smaller (labeled "$K_p/10, K_i/10$") the reaction is very slow and it takes almost 2000 intervals for the station to stop benefiting from its misbehavior.

We note that, while the analysis of Section 4.4 guarantees stability when all stations run DOC, our system is also stable when some of the stations are selfish. This is shown by the experiment of Fig. 4.11 where, after one of the stations turns selfish, the others increase their access probability to a value that ensures the selfish station does not have any gain. The system then remains stable at this point of operation.

The results show that with a larger setting of $\{K_p, K_i\}$ the system suffers from instability while with a smaller one it reacts too slowly. Hence, the proposed setting provides a good tradeoff between stability and reaction time.

4.6.6 Joining and leaving stations

To assess the effectiveness of DOC with stations joining and leaving the network, we perform the following experiment. We consider a wireless network with 5 stations, one

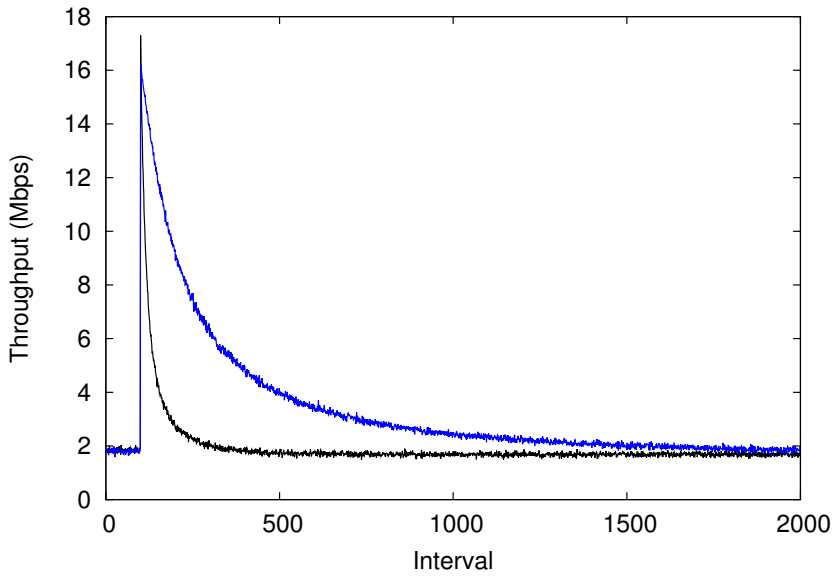


Figure 4.11: Speed of reaction provided by the parameters of the PI controller.

of which is a selfish station. After 1000 intervals, 5 additional stations join the wireless network, stay for I intervals, and then leave. The initial 5 stations stay for another 1000 intervals. The selfish station plays with a configuration $\{p_1, \bar{R}_1\}$ when there are 5 stations in the network and a configuration $\{p_2, \bar{R}_2\}$ when there are 10 stations. We obtain these configurations by performing an exhaustive search over all possible configurations and selecting the $\{p_1, \bar{R}_1\}$ and $\{p_2, \bar{R}_2\}$ values that provide the selfish station with the largest average throughput. Fig. 4.12 shows the average throughput obtained by the station with this selfish strategy compared to the throughput it would obtain if it played DOC. The results confirm that the selfish station cannot obtain any gain by deviating from DOC.

4.7 Summary

Recently proposed Distributed Opportunistic Scheduling (DOS) techniques provide throughput gains in wireless networks that do not have a centralized scheduler. One of the problems of these techniques, however, is that they are vulnerable to malicious users who may configure their parameters to obtain a greater share of the wireless resources.

In this chapter we have addressed this problem by proposing a novel algorithm that prevents such throughput gains from selfish behavior. With our approach (named DOC), upon detecting a selfish user, stations react by using a more aggressive parameter configuration that indirectly punishes the selfish station. Such an adaptive algorithm has to carefully adjust the reaction against a selfish station in order to prevent the system from becoming unstable. A key aspect of our proposal is that we use tools from *control theory*

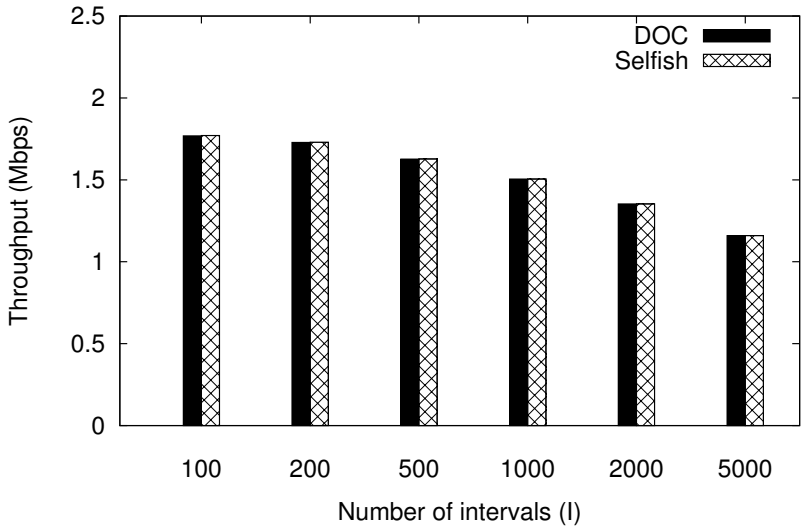


Figure 4.12: Joining and leaving the network.

combined with *game theory* to design our algorithm: by conducting a control theoretic analysis, we show that when all stations run DOC the system converges to the desired configuration, and by conducting game theoretic analysis, we show that selfish stations cannot benefit from playing a different strategy.

Chapter 5

Using DOS to Optimize Energy Efficiency

The strategies proposed in previous chapters seek to maximize throughput performance. In this chapter, in contrast, we advocate to include energy consumption to the objective utility maximization, that is, to optimize energy efficiency rather than bits per unit of time in a heterogeneous scenario, i.e., devices with different power consumption footprints. To this aim, we first present the energy consumption model based on an exhaustive experimental study of the power behavior of a set of heterogeneous wireless devices. Second, we motivate the need of considering throughput fairness when optimizing enery-efficiency; and third, we use optimal stopping theory [11] to configure DOS to maximize an energy-efficiency but throughput-fair objective.

5.1 Power consumption model

A *quantitative* treatment of the attainable energy improvements is greatly simplified by the availability of realistic and accurate energy models, also considering that *fine-grained* per-frame experimental measurements (versus coarse aggregate power consumption statistics) may be non trivial to achieve. Most of the literature works, including but not limited to [84–86, 34, 39, 87–90], ground their proposed analyses, optimizations, or algorithm/protocol designs, on the widely accepted paradigm that the energy toll may be ascribed to two components: a baseline one, plus a second one linear with (transmission/reception) air time. The specific weights are of course tailored to the interface state (transmit, receive, idle, sleep), and can be gathered by data sheets [92] or experimental measurements [81, 75].

With such a widespread acceptance, questioning the above mentioned *classical* energy model seems tough. Actually, such model makes perfectly sense if we just focus on

the network interface card consumption. But, in practice, processing in the host device drains energy as well. So, the question at stake is whether (and to what extent) there is some *energy toll in the device*, which is imputable to TX/RX processing, but which is *improperly accounted in such classical model*, e.g., because it can be neither considered (*i*) independent of the radio operation and thus (implicitly) accounted in the fixed baseline energy consumption component, nor (*ii*) strictly proportional to the traffic load in bytes, hence (implicitly) accounted in the linear air time energy cost component.

In order to characterize the power consumption of wireless devices, we have conducted an in-depth experimental investigation over a heterogeneous set of IEEE 802.11 devices. A 802.11 device is a platform very convenient for experimentation because (*i*) it is inexpensive, (*ii*) an open-source software protocol stack is available, and (*iii*) it operates over ISM radio bands, i.e., unlicensed bands. This work not only raises this question (apparently unnoticed in most prior work), but, more significantly, provides a (we believe) compelling answer, via extensive and tailored experiments providing a detailed *anatomy* of the energy consumption in the protocol stack. Even though we restrict ourselves to perform the experimental study over 802.11 equipment, the strong evidences shown in our study suggest that extending this conclusions to *any* commodity wireless platform is reasonable.

Two major findings appear to emerge. First, a *substantial* energy consumption occurs while a frame is delivered across the protocol stack, namely from the operating system to the driver to the NIC (and conversely for reception). Such "new" energy cost component, descriptively referred to as *cross-factor*, cannot be neglected; on the contrary, in some experiments it even accounts to *more than half* of the per-frame energy cost. Second, such cross-factor can be neither dealt with as an extra baseline component, nor (perhaps more surprisingly) as a cost proportional to the traffic load. Actually, this energy toll appears mostly *associated to the very fact that a frame is handled*, i.e., irrespective (to a very large extent) of the actual frame size in bytes.

Although we conducted our measurements for all the platforms in Table 5.1, in the following we only present the relevant measurements for one of them (the Soekris device with the Linux OS) to avoid redundancy and then provide a summary of the results obtained with other platforms/OS. Appendix A provides detailed explanations regarding the methodology used for the experiments and additional results for specific operations of the IEEE 802.11 protocol.

5.1.1 Energy consumption anatomy

A pre-requirement for characterization of wireless devices consists in quantifying their "baseline" power consumption, i.e., when the devices neither send nor receive traffic.

Device	Name	WiFi NIC	Mem./CPU	OS
Wireless router	Soekris net4826-48	Atheros AR5414	128MB 233MHz	Gentoo 10.0
Wireless router	Soekris net4826-48	Atheros AR5414	128MB 233MHz	OpenBSD 5.1
Wireless router	Alix 2d2	Broadcom BCM4319	256MB 500MHz	Ubuntu 10.04
Access Point	Linksys WRT54GL	Broadcom BCM4320	16MB 200MHz	OpenWRT Backfire
Smart Phone	HTC Legend	Texas Inst. WL1273	384MB +600MHz	Android 2.2
Tablet	Samsung Galaxy Note 10.1	Broadcom BCM4334	2GB +1.4GHz	Android 4.1.1
Embedded Device	Raspberry Pi	Ralink RT5370	512MB 700MHz	Raspbian Wheezy

Table 5.1: Platforms considered.

Table 5.2 reports measurements for the Soekris platform in three "baseline" configurations. Note that plugging the wireless card ("WiFi off") increases consumption by 0.29 W (+12.6%), whereas loading the driver and associating to an AP ("Idle") further increases the consumption by 0.98 W, indeed an extra 25% increment. The power consumed in the "Idle" state, named π^{id}, will be used as baseline reference in what follows.

Config.	Description	Cons. (W)
w/o card	no NIC connected	$2.29 \pm 2.2\%$
WiFi off	NIC connected driver not loaded	$2.58 \pm 2.0\%$ (+0.29)
Idle (π^{id})	NIC activated+associated to AP no RX/TX besides beacons	$3.56 \pm 1.7\%$ (+0.98)

Table 5.2: Soekris Baseline consumption profile.

5.1.1.1 Analysis of the transmission costs

Results in this section aim at characterizing the energy cost of transmissions, and providing our best effort to accurately explain and justify the relevant findings. For this reason, in the remainder of this section, results are obtained for unicast *unacknowledged* frames, so as to avoid biasing results with the cost of 802.11-specific control operations such as the reception of acknnowledgments (ACKs).

A large number of *total* device power consumption measurements have been carried out, spanning several combinations of four quantities/parameters: (i) frame size L in

the range 100 to 1500 bytes, (*ii*) modulation and coding schemes (MCS $\in$ {6, 12, 24, 48} Mbps), (*iii*) configured transmission power[1] (`txpower` $\in$ {6, 9, 12, 15} dBm), and (*iv*) frame generation rate λ^g, up to 2000 frames per second (fps).

It turns out that the most insightful way to represent such results is via a *power/airtime plot*, shown in Fig. 5.1. Such plot reports the average power consumed by the whole device, versus the percentage α^{tx} of channel airtime, computed as[2]

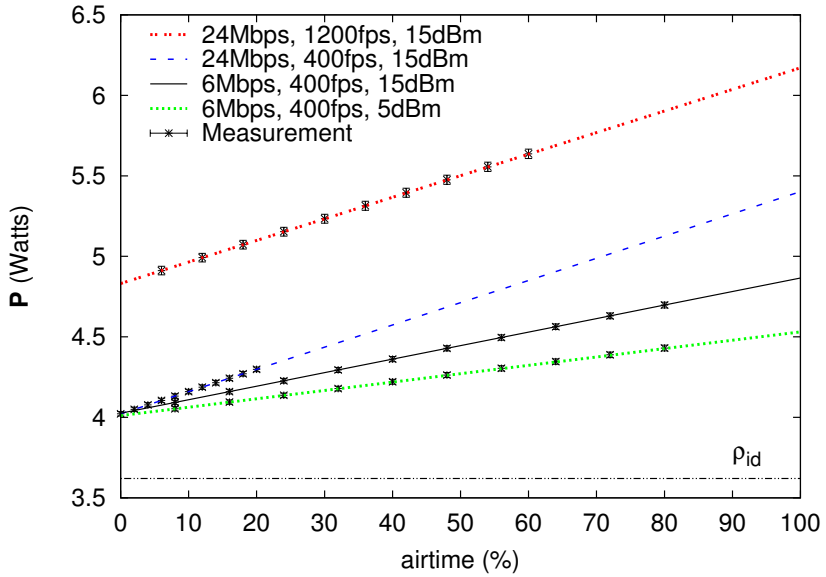


Figure 5.1: Power consumption as a function of the transmission airtime

$$\alpha^{tx} = \lambda^g T_L, \tag{5.1}$$

where λ_g is the frame generation rate, and $T_L = T_{PLCP} + (H + L)/MCS$ is the time required to transmit a frame of size L using the modulation and coding scheme MCS, duly accounting for the Physical Layer Convergence Protocol preamble T_{PLCP}, and the MAC overhead H (MAC header plus FCS). For reference purposes, the plots also report the baseline power consumption π^{id} when the target device is in "Idle" state.

Besides the *quantitative* differences among the considered platforms, these plots provide compelling evidence that the total device power consumption, denoted P, appears articulated into three main components,[3]

$$P = \pi^{id} + P^{tx} + P^{xg}(\lambda^g), \tag{5.2}$$

[1] We have selected four values within the range of allowed transmission power values, which goes from 5 to 15 dBm.

[2] The values shown in the figures are the result of applying a simple linear regression to the measurements and computing their standard asymptotic error [107].

[3] The good match between the experimental figures and equation 5.2 is confirmed in Fig. 5.1, in which the values predicted by the equation are plotted using lines.

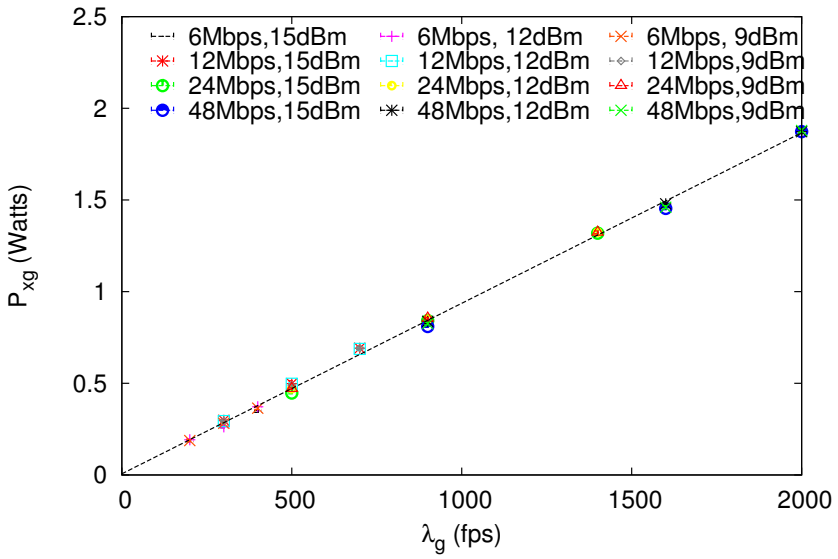


Figure 5.2: Relation between $P^{xg}(\lambda^g)$ and λ^g.

where:

- The first component, π^{id}, is the (platform-specific) baseline power consumption.
- The second component, P^{tx}, is the classical one in traditional energy consumption models, which linearly grows with the airtime percentage α^{tx}, i.e., $P^{tx} = \pi^{tx}\alpha^{tx}$. The slope π^{tx} depends on the target platform and on the radio transmission parameters MCS and `txpower`: the greater the MCS and/or the `txpower`, the greater the slope.
- The third component, $P^{xg}(\lambda^g)$, accounts for the fact that the above linear trend does *not* start from the baseline power consumption level π^{id}, but rather *starts from a relatively large positive offset* (e.g., in the Soekris case, +12% and +35% increment over the baseline level π^{id} for 400 and 1200 fps, respectively); offset which is *not accounted by classical energy models* [84–86, 34, 39, 87–90]. Moreover, Fig. 5.1 suggests that such component depends only on the frame generation rate λ^g.

To more closely investigate the nature of such emerging power consumption offset $P^{xg}(\lambda^g)$, Fig. 5.2 plots its value obtained from several measurements taken for different configuration of the NIC parameters (MCS, `txpower`) over the Soekris platform (results are qualitatively analogous for the other two platforms). The plot clearly shows that $P^{xg}(\lambda^g)$ is *proportional* to the frame generation rate λ_g, whereas it is practically *independent of* the frame size or the radio settings.

Thus, if we denote with $\gamma^{xg} = P^{xg}(\lambda^g)/\lambda^g$ the proportionality constant, it appears that γ^{xg} is the *enery toll associated to the processing of each individual frame*, irrespective

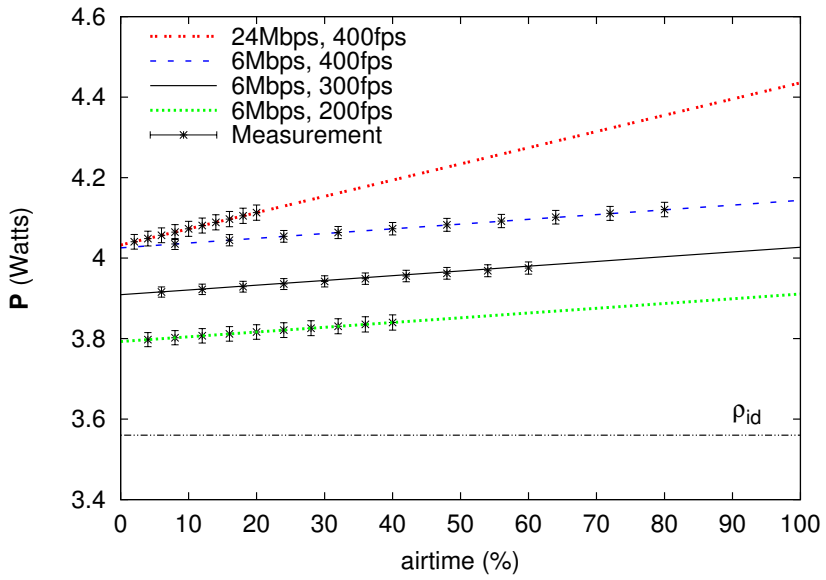


Figure 5.3: Power consumed by (unacknowledged) reception versus airtime.

of its size or radio transmission parameters. Note that this energy toll is *not* associated to protocol operations specific of the technology selected for experimentation such as RTS/CTS or ACKs, indeed disabled in such experiments. We call this per-frame energy toll the *cross-factor*.

5.1.1.2 Analysis of the reception costs

The analysis of the power consumption of the device while *receiving* frames is somewhat dual to that carried out in depth for the previous transmission case, hence may be dealt with much faster. We use the same configurations of MCS and `txpower` as in Section 5.1.1.1 (ACK disabled as well), with different combinations of the frame length L and frame *reception* rate λ^r. The resulting *power/airtime* plot is shown in Fig. 5.3 (Soekris device), airtime now given by $\alpha^{rx} = \lambda^r T_L$. The `txpower` parameter is not shown, as it does not affect the power consumption (as indeed well known from [82]).

Fig. 5.3 exhibits the same qualitative pattern found in the transmission scenario. The increment of the power consumption over π^{id} is composed of two components: a first one linear with the airtime and accounting for the power required to receive frames, π^{rx} (indeed in line with traditional energy models), and a second one proportional to the number of frames received and accounting for the cross-factor energy toll, $\pi^{xr}(\lambda^r)$. The total power consumption at the receiving side is thus:

$$P = \pi^{id} + P^{rx} + P^{xr}(\lambda^r) = \pi^{id} + \pi^{rx}\alpha^{rx} + \lambda^r \gamma^{xr}, \tag{5.3}$$

where γ^{xr} is the cross-factor in reception, i.e., the per-packet processing toll to deliver the received frame across the protocol stack, and $\pi^{rx}\alpha^{rx}$ is the traditional reception cost proportional to the airtime. Again, Fig. 5.3 confirms that results from the above equation (lines) closely match the experimental measurements (symbols).

To analyze the impact of the transmissions addressed to another station, we configured a communication between two nodes and measured the energy consumption at a third node that was in the transmission range of this communication. We observed that the energy consumed by this node was the same as if the medium was idle, which confirms that transmissions addressed to other stations practically do not consume energy. This is in agreement with our previous results: according to (5.3), the energy cost of listening to the PLCP plus headers is only 38 μJ/frame (for 6 Mbps MCS), which has practically no impact on the overall consumption.

5.1.1.3 Collisions and other transmissions

We now analyze the impact on the energy consumption in reception when the medium is occupied by collisions or by transmissions addressed to another device (i.e., to another MAC address).

For this purpose, we configured a communication between two nodes and set up another node to act as interferer. The sender and the interferer were configured with the carrier sense threshold at the highest value, which practically results in no carrier sensing, and the interferer used the lowest values for the CW, $SIFS$[4] and MCS parameters while deactivating the use of ACKs, resulting in practically continuous transmission. To control the amount of time the interferer was sending data (i.e., the 'interference rate'), we used the `quiet element`[5] option to silence the interface for a given amount of time every beacon period. With this setting, the interferer transmits continuously during a long period, emulating thus the typical behavior of collisions with carrier-sensing which always affect frame transmissions from the beginning.[6] Prior to our measurements, we performed extensive tests using different `txpower` configurations and varying the relative position of the devices, to find a setting in which simultaneous transmissions resulted in all frames being lost (i.e., no capture effect).

Table 5.3 presents the measured power consumption for different sending and interference rates. We observe that (i) the power consumed in reception depends exclusively

[4]The CW (Contention Window) parameter tunes the random backoff procedure specific of IEEE 802.11. Note that it can be configured to be proportional to the p_i's [108] used in our DOS system. $SIFS$ is an inter-frame space, also specific of IEEE 802.11, during which the station does not transmit/receive frames.

[5]The `quiet element` is a data structure used in IEEE 802.11h to force stations to remain silent, i.e., transmit nothing, to assist Access Points in making channel measurements without interference.

[6]Note that, with hidden nodes, interfering frames could also arrive at a later point of the frame transmission. However, the measurements of [109] show that such interfering frames, which do not affect the preamble, need to have a very high power to cause a transmission error.

on the traffic actually received (see, e.g., when the interference rate goes from 0% to 50%), and (*ii*) collisions have the same impact as an idle medium (e.g., the cases with 100% interference rate coincides with ρ_{id}). Based on this, we conclude that collisions have no practical impact on the energy consumption at the receiver (this is confirmed by the results of the model included in the table).

Table 5.3: Impact of collisions on reception.

Frames/s sent	Interference rate	Frames/s received	Measured power	Model
100 fps	0%	100 fps	3.67 W	3.68 W
200 fps	50%	100 fps	3.67 W	3.68 W
200 fps	100%	0 fps	3.56 W	3.56 W
200 fps	0%	200 fps	3.80 W	3.81 W
400 fps	50%	200 fps	3.80 W	3.81 W
400 fps	100%	400 fps	3.56 W	3.56 W

To analyze the impact of the transmissions addressed to another station, we configured a communication between two nodes and measured the energy consumption at a third node that was in the transmission range of this communication. We observed that the energy consumed by this node was the same as if the medium was idle, which shows that transmissions addressed to other stations practically do not consume energy.

5.1.2 Other devices

The results provided in the previous sections have been obtained for one particular hardware platform (the Soekris device) running one specific OS (Linux Gentoo 10.0). In order to verify that the behavior observed in those sections is not specific of the chosen reference device, OS or WLAN band/card/PHY, we repeated all the experiments for each of the platforms listed in Table 5.1 (results not included here to avoid redundancy). These experiments show that, albeit there are obviously some quantitative differences in the energy consumed by each device, all the devices under study show the same qualitative behavior.

In addition to their qualitative behavior, it is also interesting to analyze the quantitative differences between the energy components of the different devices. In particular, one of the key results drawn from the previous section is that the cross-factor has a substantial impact over the total energy consumed by a frame. In order to gain insight onto the weight of the cross-factor in different platforms and OS, Fig. 5.4 depicts, for each of the devices of Table 5.1, how the per-frame energy consumption is decomposed into the following two components: the *cross-factor* component (P^{xg} or P^{xr}) and the transmission/reception component (P^{tx} or P^{rx}). Results are provided for $MCS = 48Mbps$, the default `txpower` configuration and two different packet lengths (100 and 1500 bytes,

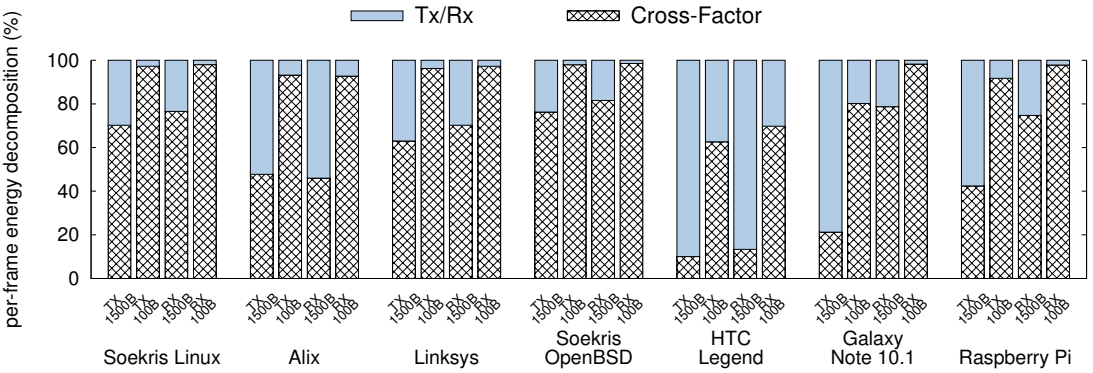


Figure 5.4: Fraction of the cross-factor over the total per-frame energy consumption cost.

respectively).[7] For those devices that do not implement CPU scaling, we used the default CPU frequency configuration, while for those that implement it we used the largest CPU frequency.[8]

From the results obtained, we observe that the *cross-factor* has a very significant weight in the vast majority of the devices and configurations studied, as it accounts for *more than 50%* of the per-frame energy consumption in most cases. Only for the cases of transmitting large packets with the Galaxy Note, and transmitting and receiving large packets with the HTC Legend device, the *cross-factor* has a less substantial (but still significant) weight; in particular, in those cases the *cross-factor* accounts for approximately 20% and 10% of the energy consumed by a frame, respectively.

The results obtained also show the impact of the operating system on the *cross-factor*. Indeed, since the *cross-factor* is caused by the frame processing at the different layers of the protocol stack, it depends on the operating system implementation. To gain insight into the impact of the OS, Fig. 5.4 includes two devices that rely on the same hardware platform (Soekris) but run two different OS (Linux and OpenBSD, respectively). The measurements obtained confirm that the qualitative behavior with both operating systems is the same, and show that the *cross-factor* is of the same order (although, as expected, the specific values are slightly different as a result of the different protocol stack implementation).

From the results reported above, we draw the following conclusions: (i) the behavior observed in the previous section is not specific form the Soekris device but is generalized to

[7]Since transmission/reception component is proportional to transmission/reception time, it heavily depends on the packet length; thus, by evaluating very large (1500 bytes) and very small (100 bytes) packets, we cover the the two extreme cases for the weight of the *cross-factor*.

[8]The HTC Legend and Galaxy Node devices implement by default CPU scaling, which adaptively sets the CPU frequency based on the CPU load. We observed, however, that this leads to quite unpredictable consumption and thus disabled this function in our experiments, setting the CPU frequency to a fixed value. Table A.1 in Appendix A shows the impact of different CPU frequency values onto energy consumption.

all the devices under study, which includes most of the types of 802.11 devices of practical interest; and (*ii*) the weight of the *cross-factor* on the per-frame energy consumption is significant in all cases, even though as expected it is smaller for those devices that rely on a more energy efficient design, such as the smart phone and the tablet.

These results thus confirm that the conclusions drawn above respond to *general energy consumption patterns* of all 802.11 wireless devices (or at least a very wide range), and thus we assume this behavior could be extended to all *real-life* commodity wireless devices including the DOS-capable stations covered in this thesis.

5.1.3 Energy consumption model

The complete model

Based on the results obtained in the previous section, we now build a *complete model* for the power consumption of wireless devices. Summarizing our findings, we have that the power consumed by a wireless device i consists of the following components: (*i*) the idle consumption, π^{id}, (*ii*) the cross-factor for the packets generated by the application, P^{xg}, (*iii*) the power required to transmit them, (*iv*) the power spent in receiving frames, P^{rx}, and (*v*) the cross-factor for the received frames, P^{xr}[9]:

$$P_i = \pi_i^{id} + P_i^{tx} + P_i^{xg} + P_i^{rx} + P_i^{xr}. \tag{5.4}$$

By taking into account the transmission airtime percentage, α^{tx}, and the reception airtime percentage, α^{rx}, the above equation can be rewritten as:

$$P_i = \pi_i^{id} + \pi_i^{tx}\alpha_i^{tx} + \pi_i^{rx}\alpha_i^{rx} + \gamma_i^{xg}\lambda_i^{g} + \gamma_i^{xr}\lambda_i^{r}. \tag{5.5}$$

The above expression gives the model for the power consumption of a wireless device proposed for this chapter. As already mentioned in the previous section, the key difference between the above model and the 'traditional' one used in many previous works [84–86, 34, 39, 87–90] is that the traditional model only includes the first three components (namely ρ_{id}, $\pi^{tx}\alpha^{tx}$ and $\pi^{rx}\alpha^{rx}$) while our model adds to these three components two additional ones ($\gamma^{xr}\lambda^{r}$ and $\gamma^{xg}\lambda^{g}$). As shown by our measurements, these two additional components account for a very significant portion of the power consumption, which renders the traditional model highly inaccurate.

Out of the 9 variables in (5.5), 5 are constant parameters that depend on the device and the configuration of its communication parameters (π^{id}, π^{tx}, π^{rx}, γ^{xr} and γ^{xg}),

[9] Although not shown here, we show in Appendix A that the control operations specific of the MAC protocol do not pay the cost of the cross-factor as they are emitted straight from the hardware NIC.

while the other 4 parameters are variables that depend on the number of stations in the wireless network and their traffic generation behavior (α^{tx}, α^{rx}, λ^r and λ^g). In Appendix A (Table A.1) we characterize the 5 constant parameters that determine the power consumption of the considered 802.11 devices for different values of MCS and `txpower`.

5.1.4 Validation

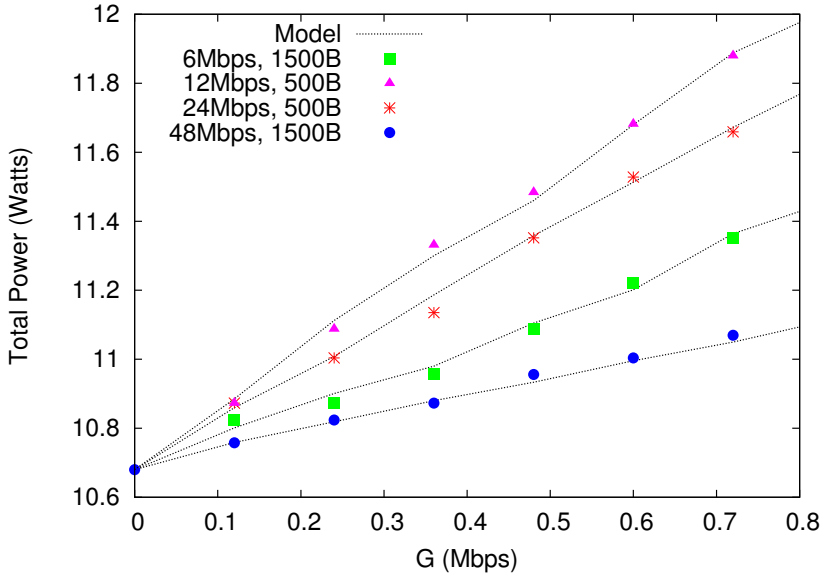


Figure 5.5: Model validation with multiple stations.

To validate our model in a general scenario with multiple sending and receiving stations, we consider a wireless network with one IEEE 802.11 AP and three stations. Each station generates unicast traffic to the AP at a rate G, while the AP sends unicast traffic at the same rate G to each station. To apply the model of (5.5), we need to obtain the parameters α^{tx}, α^{rx}, λ^r and λ^g. These can be obtained from typical statistics recorded by the wireless driver.

We compare the results given by our model against those obtained from measurements. Fig. 5.5 depicts these results for various combinations of L and MCS, sweeping along different traffic generation rates G in the x axis. We conclude from the figure that the proposed model is able to accurately predict the power consumption in a general scenario.

5.2 Balancing Energy Efficiency and Throughput Fairness

Along with a tremendous growth in number of mobile computing devices, we currently witness an increasing diversity in the energy profile of those devices. Given the existing

heterogeneity, performing a *good* allocation of wireless resources among devices to improve energy efficiency is challenging; indeed, the relation of energy efficiency with throughput optimization has received little attention so far, [110] being one of the few works to analyze these various tradeoffs in wireless networking.

To solve this, we first propose a criterion to balance between the lack of energy considerations of a throughput-based allocation and the extreme unfairness of a purely energy-based allocation. Then, we show how to apply this criterion to DOS in both homogeneous and heterogeneous networks. Throughout the chapter we will denote with η the energy efficiency of the network, i.e.,

$$\eta = \frac{throughput}{power} \tag{5.6}$$

5.2.1 Motivation for an energy efficient yet fair channel access

For the case of heterogeneous scenarios, where different stations have different power consumption figures, it is not trivial to define the performance figure to optimize, as utilizing a naïve approach towards optimization of the network-wide energy consumption might result in the starvation of devices. In the following we illustrate why these scenarios constitute a different and more challenging case to tackle.

Let us consider a toy scenario with $N = 10$ stations divided into two groups of 5 stations each: G_1 and G_2 and average normalized SNR $\rho = 1$ for all links. Stations in G_1 and G_2 are modeled after the device parameters "Soekris" and "Alix" from Table 5.4, respectively. We denote with $\{p_1, \bar{R}_1\}$ and $\{p_2, \bar{R}_2\}$ the access probability and rate threshold configuration used by G_1 (G_2), and use two different strategies to configure these parameters:

- Strategy I: We set $p_1 = p_2$ and $\bar{R}_1 = \bar{R}_2$ in order to have a fair share of the wireless resources, and perform an exhaustive search over the parameters space to choose the value that maximizes throughput.
- Strategy II: We let each group's configuration diverge, and perform an exhaustive search over the parameters space to find the configuration that maximizes the energy efficiency η of the network.

For the first strategy the resulting optimal configuration is $\{p_1 = p_2 = 0.1, \bar{R} = 8.98$ Mbps$\}$. For the second strategy, the resulting configuration is $\{p_1 = 0.01, \bar{R}_1 = 13.27$ Mbps$\}$ and $\{p_2 = 0.19, \bar{R}_2 = 12.27$ Mbps$\}$. We report the results obtained in Fig. 5.6, with the following main conclusions:

- The first strategy provides a bandwidth-fair allocation where both stations receive the same throughput, while the overall energy efficiency is 0.11 MbpJ.

Table 5.4: Power consumption parametrization for a set of devices in Table A.1

Device	Soekris	Alix	Linksys
π_{tx} (W)	0.86	0.40	0.97
π_{id} (W)	3.56	3.68	2.73
γ_{xg} (mJ)	0.93	0.11	0.46

- The second strategy, on the other hand, results in an energy-efficiency improvement of approximately 32%. However, the resulting throughput allocation is extremely unfair, as stations in G_1 are practically starved.

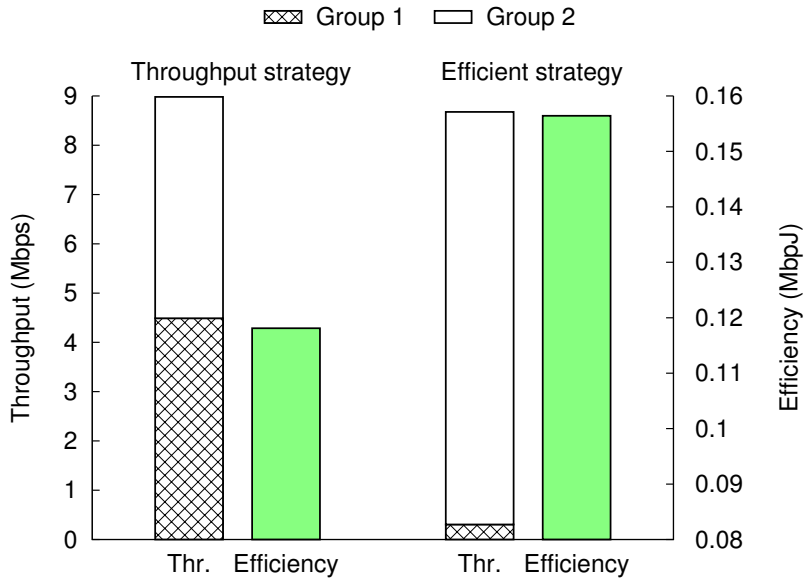


Figure 5.6: Performance of a simple network for different configuration strategies.

The fact that the most energy-efficient allocation (Strategy II) is obtained using an extremely unfair allocation is caused by the random-based channel access scheme, as choking one device will prevent the energy wastage caused by collisions. The price to pay for increasing the efficiency is then to introduce unfairness.

This simple scenario serves to illustrate the risks of using a naïve strategy to optimize the overall energy efficiency. On the other hand, it is clear that the use of throughput-only allocation criteria, while resulting in throughput-fair allocations, do not consider energy efficiency at all as they do not take into account the power consumption profile of the stations.

Based on the above, we claim that a tradeoff between energy efficiency maximization and throughput fairness is needed. In the following we discuss our proposed criterion to define this tradeoff.[10]

[10]Note that we have only considered power consumption figures, and not parameters such as, e.g., the

5.2.2 A criterion for energy efficient and fair channel access

The use of overall energy efficiency figures, as we have seen in the previous section, is not well suited to properly address general (i.e., heterogeneous) scenarios. The use of throughput-based approaches, on the other hand, does not consider the impact of the different power consumption parameters and therefore may result in energy wastage. We argue that a tradeoff between these two is needed.

In order to define a tradeoff between these two different optimization objectives, we first define the *per-station* energy efficiency η_i as the ratio between the throughput obtained and the power consumed by a given station i:

$$\eta_i = \frac{throughput_i}{power_i}$$

Note that η_i provides the throughput the station i is successfully transmitting weighted by the energy the station has to spend and, therefore, it partially takes into account if a station is being choked. This way, for our toy example in the previous subsection, the resulting values for the first configuration strategy (throughput-fair strategy) are $\eta = \{0.09, 0.15\}$ MbpJ, while for the case of the second strategy the values are $\eta = \{0.01, 0.22\}$ MbpJ.

Based on these η_i variables, our challenge is to define an appropriate criterion for their configuration. To this aim, note that we have a two-fold objective: on one hand, we want to maximize the overall efficiency η in the network; on the other hand, we want to preserve some degree of fairness between the η_i's, thus avoiding that any station is starved. In Chapter 3 and 4 we tackle a similar challenge, in these cases to balance between a rate allocation that maximizes total throughput and another where the capacity of the channel is equally shared. In order to solve this tradeoff, we (and many other works in similar allocation problems) use a proportional fair (PF) criterion [16].[11]

In this chapter, following the previous works of [112–114] we advocate for the use of a similar criterion for a wireless network with heterogeneous stations. This way, we propose to use the energy-efficiency proportional fairness (EF) criterion, based on the maximization of the sum of the per-station energy efficiency, i.e.,

$$EF \iff \max \sum \log(\eta_i) \tag{5.7}$$

remaining battery capacity. Although such battery parameters have been considered before in energy-related scenarios (e.g., in [111]), they are not well suited for the scenarios that we envision. Indeed, the approach that we propose provides an incentive to energy-efficient devices by favoring them over inefficient ones. In contrast, a solution that favored battery constrained devices would incentivize battery limited devices which would harm the overall performance. Following this reasoning, in this thesis we only focus on the energy efficiency of the different wireless devices implementing the protocol.

[11] See Chapter 1 for a brief introduction of this criterion.

To illustrate why the use of the EF criterion prevents extremely unfair allocations while supporting energy-efficient configurations, we consider the same heterogeneous scenario used before with $N = 10$ stations divided into two different groups of 5 stations, each one modeled after the power consumption figures of devices "Soekris" and "Alix" from Table 5.4. In order to analyze different configurations of the $\{p, \bar{R}\}$, we set $p_1 = Kp_2$ and $\bar{R}_1 = M\bar{R}_2$ with K ranging from 0.4 to 1.5 and M ranging from 0 to 4, and for each $\{K, M\}$ value we perform a sweep on the $\{p_1, \bar{R}_1\}$ parameters to obtain the configuration that maximizes the EF. For each resulting configuration we compute the throughput of each station, the overall energy efficiency η and the EF value given by (5.7). We plot the results of this experiment as a function of K, and for different values of M (including the M that maximizes the EF, namely, "optimal"[12]), in Fig. 5.7, and can be summarized as follows:

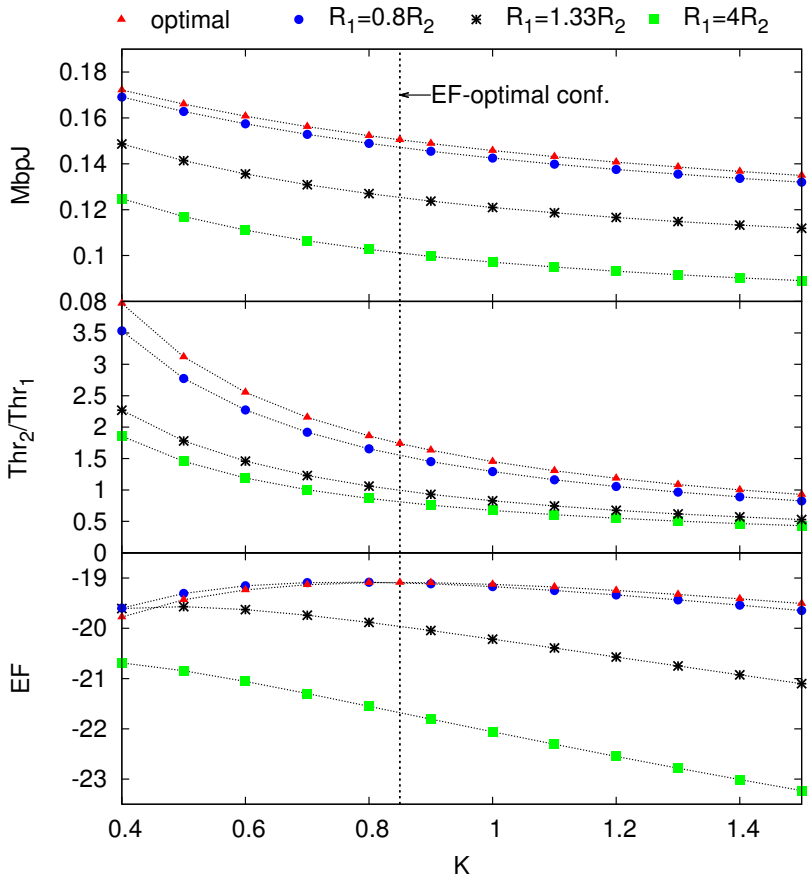


Figure 5.7: Performance of a simple network for different $\{p, \bar{R}\}$.

- Low p_1/p_2 ratios increase the overall efficiency η, but lead to the starvation of G_1,

[12]Note that, for the sake of readability, throughout the thesis we use EF to refer *both* to the quantity $\sum \log(\eta_i)$ resulting from a particular configuration, and to the criterion that maximizes this value. The distinction will be clear based on the context.

as can be seen from the Thr_2/Thr_1 ratio. This is the result that we have seen in the previous section, namely, that the most energy-efficient configuration is the one that chokes the least efficient device, the "Soekris" in this case.

- However, the value of EF is not maximized for such extremely unfair allocations, but instead the maximum is reached when $k \approx 0.85$. From this point on, the relative increase in η_1 (η_2) is not compensated by the relative decrease of η_2 (η_1) and, therefore, the allocation is not EF-optimal.

In our toy example, the η-optimal allocation is given by the duple $p = \{0.05, 0.13\}$ and $\bar{R} = \{13.27, 9.87\}$ Mbps, which provides an overall efficiency $\eta = 0.17$ MbpJ and a throughput allocation $Thr = \{0.35, 1.40\}$ Mbps for each station of both groups[13]. On the other hand, the EF-optimal configuration $p = \{0.09, 0.106\}$ and $\bar{R} = \{13.27, 9.87\}$ Mbps results in the following performance figures: $\eta = 0.15$ MbpJ and $Thr = \{0.63, 1.09\}$ Mbps. For this case, then, the EF-optimal configuration exchanges a 13% reduction in the overall efficiency in order to improve throughput fairness[14] from 0.73 to 0.93.

According to these results, we conclude that the EF-criterion can be used to balance adequately the tradeoff between a fair throughput allocation and an energy-efficiency configuration. Moreover, although the rest of the chapter is devoted to the case of DOS networks, indeed the topic of this thesis, this criterion could be applied to any scenario with heterogeneous devices.

5.3 An Energy-efficient Fair strategy for DOS

The objective of this section is therefore to find the configuration strategy that maximizes the EF for a DOS system, i.e., that finds a good balance allocation between throughput fairness and energy efficiency. With the above, we can formulate our problem as the following optimization problem

$$\underset{\mathbf{p},\bar{\mathbf{R}}}{\text{maximize}} \quad \sum \log(\eta_i). \tag{5.8}$$

We describe the average energy consumption of a DOS station i as

$$e_i = \quad p_e E_{e,i} + p_{s,i} E_{s,i} + p_{s,\neg i} E_{s,\neg i} + p_{c,i} E_{c,i} + p_{c,\neg i} E_{c,\neg i} \tag{5.9}$$

where,

[13]Note that we have constrained the ratio p_1/p_2 to be $0.4 < \frac{p_1}{p_2} < 1.5$ and that we could keep increasing the overall efficiency by penalizing fairness further.

[14]According to Jain's fairness index [104].

- $p_e E_{e,i}$ is the energy consumed due to empty slots.
- $p_{s,i} E_{s,i}$ is the energy required to transmit a frame of duration $\mathcal{T}$ times the probability of such event occurring.
- $p_{s,\neg i} E_{s,\neg i}$ is the energy spent during other stations' transmissions.
- $p_{c,i} E_{c,i}$ is the energy invested during a collision of station i's transmission times the probability of a collision of a station's i frame in the network.
- $p_{c,\neg i} E_{c,\neg i}$ is the energy consumed due to collisions where station i is not involved.

We can compute the previous probabilities as

$$\begin{aligned} p_e &= \prod_j (1 - p_j) \\ p_{s,i} &= p_i \prod_{j \neq i} (1 - p_j) \\ p_{s,\neg i} &= \sum_{j \neq i} p_j \prod_{k \neq j} (1 - p_k) \\ p_{c,i} &= p_i - p_i \prod_{j \neq i} (1 - p_j) \\ p_{c,\neg i} &= 1 - p_i - p_e - p_{s,\neg i} \end{aligned} \tag{5.10}$$

Now, based on the power consumption model for wireless stations devised in Section 5.1, we compute the energy consumed on each events as[15]

$$\begin{aligned} E_{e,i} &= \pi_i^{id} \tau \\ E_{s,i} &= Prob(R_i(\theta) < \bar{R}_i) \left((\tau + \mathcal{T})(\pi_i^{id} + \pi_i^{tx}) + \gamma_i^{xg} \right) + \\ &\quad + Prob(R_i(\theta) \geq \bar{R}_i)(\pi_i^{id} + \pi_i^{tx})\tau \\ E_{s,\neg i} &= \sum_j \frac{p_{s,j}}{p_s} (Prob(R_j(\theta) < \bar{R}_j)(\tau + \mathcal{T})(\pi_i^{id}) + \\ &\quad + Prob(R_j(\theta) \geq \bar{R}_j)(\pi_i^{id})\tau) \\ E_{c,i} &= (\pi_i^{id} + \pi_i^{tx})\tau \\ E_{c,\neg i} &= \pi_i^{id} \tau \end{aligned} \tag{5.11}$$

where π_i^{id} and π_i^{tx} represent the base and transmission power consumption, respectively, and γ_i^{xg} is the energy consumption (in Joules) due to the software processing required to generate a transmission, the *cross-factor*, as explained in Section 5.1. These values

[15] We assume that π^{tx} is the power consumption figure used for probing the channel, as well as for transmit regular data.

characterize the power behavior of each station and we assume they are fixed regardless the rate in use.[16] If we rearrange some terms, we can compute the average energy consumption of station i as

$$\begin{aligned} e_i = & \pi_i^{id}\tau \\ & + p_i\pi_i^{tx}\tau \\ & + \sum_{j\neq i} p_{s,j} Prob(R_j(\theta) \geq \bar{R}_j)(\pi_i^{id}\mathcal{T}) \\ & + p_{s,i} Prob(R(\theta) \geq \bar{R})(\mathcal{T}(\pi_i^{id} + \pi_i^{tx}) + \gamma_{xg}) \end{aligned} \tag{5.12}$$

and the average power consumption of station i as

$$P_i = \frac{e_i}{\sum_j p_{s,j}T_j + (1-p_s)\tau} \tag{5.13}$$

where T_i can be computed as

$$T_i = Prob(R_i(\theta) < \bar{R}_i)\tau + Prob(R_i(\theta) \geq \bar{R}_i)(\mathcal{T} + \tau).$$

Considering all the terms above and the throughput model devised in Chapter 3, we can compute the energy efficiency of station i as

$$\eta_i = \frac{r_i}{e_i} = \frac{p_{s,i}l_i}{e_i} \tag{5.14}$$

where

$$l_i = \int_{\bar{R}_i}^{\infty} rT_i f_{R_i}(r)dr$$

and $f_{R_i}(r)$ is the pdf of $R_i(\theta)$, the distribution of the available channel rate.

5.3.1 Homogeneous Stations

We first consider the case of homogeneous stations, that is, all the stations in the DOS network share similar power consumption profiles and channel quality links. In these scenarios all the stations receive the same allocation of resources and therefore maximizing EF optimizes as well the overall energy efficiency, η. Thus, in this homogeneous case, the tunable set of parameters of the system is reduced to just two: $\mathbf{p} = \{p\}$ and $\bar{\mathbf{R}} = \{\bar{R}\}$, i.e., $p_i = p,\ \bar{R}_i = \bar{R}\ \ \forall i$, and we focus on maximizing overall energy efficiency.

The configuration strategy we design is the following. First, all stations configure the same access probability $p = 1/N$. The rationale behind this is that this configuration

[16] The complete characterization for all devices in Table 5.1 is shown in Appendix A (Table A.1). The variations of π^{tx} with respect to the rate used to transmit are very moderated for practically all the devices, thus, we assume is constant in the way we model it.

gives an optimal success probability [96] and it is fair, indeed . Second, given the **p** configuration proposed before, we look for the transmission rate threshold policy that maximizes overall energy efficiency. To this aim we use optimal stopping theory [11], a technique used in [7] to maximize overall throughput performance. Let us define the problem as a maximum rate of return problem:

$$\max_{N\in\mathcal{N}} \frac{E[Y_N]}{E[\psi_N]} = \lambda, \tag{5.15}$$

where Y_N is the number of bits that can be delivered if using strategy N, and ψ_N is the energy required to do so. We can compute $Y_N = \eta_N(\theta)(\mathcal{T}\pi_{tx} + \gamma_{xg})$, where $\eta_N(\theta)$ is the instantaneous ratio of bits per joule required to transmit during $\mathcal{T}$ at time θ if the station decides to transmit using the stopping rule N. Note that λ is in bits/joule as well. $E[\psi_N]$ is the expected energy consumption required for a station to transmit a frame, i.e., to use its transmission opportunity using stopping rule N. To be pragmatic, and following the directions of [11], we restrict the set of stopping rules to those that take at least one observation and that have a bounded consumption:

$$\mathcal{N} = \{N : N \geq 1, E[\psi_N] < \infty\} \tag{5.16}$$

The problem can be transformed into an ordinary stopping rule problem: $Y_N - \lambda\psi_N$ for some λ. According to Th.1 in Ch.6 in [11], if we find such λ so that the optimal expected reward is zero, i.e. $V^* \equiv \sup_{N^*\in\mathcal{N}} E[Y_N - \lambda\psi_N] = 0$, then N^* is the optimal stopping rule of the original problem and $\lambda^* : V^*(\lambda^*) = 0$ is the maximum rate of return, i.e., the maximum energy efficiency.

Let's assume that, for any given λ, it exists an $N(\lambda) \in \mathcal{N}$ such that $E[Y_N - \lambda\psi_N]$ is maximized. Then,

$$V(\lambda) = \sup_{N\in\mathcal{N}} \{E[Y_N] - \lambda E[\psi_N]\} = E\left[Y_{N(\lambda)}\right] - \lambda E\left[\psi_{N(\lambda)}\right]. \tag{5.17}$$

It can be easily proved that $V(\lambda)$ is decreasing and convex. Now, following [11], we define our stopping rule as

$$N(\lambda) = \min\{N \geq 1 : \eta(\theta)(\gamma_{xg} + \mathcal{T}\pi_{tx}) \geq V^*(\lambda) + \lambda(\gamma_{xg} + \mathcal{T}\pi_{tx}) \tag{5.18}$$

where $\pi_{tx} = \pi_{tx} + \sum_i \pi_{id}$.[17] It can also be shown that $V^*(\lambda)$ satisfies the *optimality equation* (see [11] for more details):

$$V^*(\lambda) = E[\max\{\eta(\theta)(\gamma_{xg} + \mathcal{T}\pi_{tx}) - \lambda\psi(\theta), V^*(\lambda) - \lambda e_{cp}\tau\}] \tag{5.19}$$

[17] for nomenclature convenience, we remove the subindex i for the homogeneous scenarios, i.e, $\pi_i^{tx} = \pi_{tx}$.

where e_{cp} is the average energy consumption during a channel probing. e_{cp} can be computed as

$$e_{cp} = \sum_i E_{c,i} K_{c,i} + \sum_i E_{c,\neg i} K_{c,\neg i} + \sum_i E_{e,i} K_e + \tau(\sum_i \pi_{id} + \pi_{tx}) \tag{5.20}$$

where $K_{c,i}$, $K_{c,\neg i}$ and K_e are the average number of slots with a collision involving i, a collision not involving i and empty slots, respectively. Note that we can compute these averages as

$$K_{c,i} = \frac{\left(\frac{1}{p_s} - 1\right) p_{c,i}}{1 - p_s}, \quad K_{c,\neg i} = \frac{\left(\frac{1}{p_s} - 1\right) p_{c,\neg i}}{1 - p_s}, \quad K_e = \frac{\left(\frac{1}{p_s} - 1\right) p_e}{1 - p_s}. \tag{5.21}$$

If we set $V^*(\lambda^*) = 0$, then

$$\lambda^* \frac{\tau e_{cp}}{\gamma_{xg} + \mathcal{T}\pi_{tx}} = E\left[\eta(\theta) - \lambda^*\right]^+ \tag{5.22}$$

and the optimal rule is a pure threshold policy such that

$$N^*(\lambda^*) = \min\{\theta \geq 1 : \eta(\theta) \geq \lambda^*\} \tag{5.23}$$

Note that, $\eta(\theta) = R(\theta)\frac{(\mathcal{T})}{\gamma_{xg}+\mathcal{T}\pi_{tx}}$ and, similarly, $\lambda^* = \bar{R}\frac{(\mathcal{T})}{\gamma_{xg}+\mathcal{T}\pi_{tx}}$. Thus, the optimal threshold can be obtained by solving the following fixed point equation:

$$\bar{R}^* \frac{\tau e_{cp}}{(\gamma_{xg} + \mathcal{T}\pi_{tx})} = E\left[R(\theta) - \bar{R}^*\right]^+ \tag{5.24}$$

This way each station can compute its optimal channel access probability, p_i, and transmission rate threshold, $\bar{R}_i$, with the only requirement of inferring the number of contending stations. Note that the adaptive algorithm proposed in Chapter 3 to tune the $\bar{R}_i$ in case of using a proportional fair optimization could be easily modified to account for this energy efficient configuration of the thresholds.

5.3.2 Heterogeneous Stations

A scenario with a heterogeneous set of devices, where an EF-maximal configuration would presumably allocate resources differently depending on the power profile of each device, is indeed more challenging. Note that the complexity of the problem is much larger as the utility $\sum_i \log(\eta_i)$ has non-linear relationships with a larger set of optimization parameters $\mathbf{p} = \{p_1, \ldots, p_N\}$ and $\bar{\mathbf{R}} = \{\bar{R}_1, \ldots, \bar{R}_N\}$.

First we observe that, although the EF performance in our toy example from Sec-

tion 5.2 (in particular Fig. 5.7) is maximized for a p_1/p_2[18]≈ 0.85, this optimal EF value is very approximated in $p_1/p_2 = 1$, i.e., same channel access probability for all stations. Based on this observation, we propose the following heuristic strategy. First, similarly as for the homogeneous case, all stations configure the same access probability $p_i = 1/N$. Then, each device i configures its own threshold, independently of the other device's power profiles, as the threshold that maximizes the overall energy efficiency in a homogeneous scenario with N stations featuring i's power profile, i.e., following the policy devised in (5.23) and (5.24). In this way, the only variable that the stations need to know is the number of contending stations, and the configuration is computed distributively by each device of the DOS network. The rationale behind this algorithm is that energy-hungry stations will configure a relatively larger threshold than the energy-efficient stations while we still keep a fair channel probing among all stations.

5.4 Performance Evaluation

We evaluate the performance evaluation of our strategy by means of simulations. We use the same simulator as in previous chapters where the available rate in the wireless channel is modeled after $R(h) = Wlog_2(1 + \rho|h|^2)$ bits/s, where W is the channel bandwidth, ρ is the normalized average Signal to Noise Ratio and h is the random gain of Rayleigh fading. We fixed $\rho = 1$ in all the experiments of this chapter, but we tested different link qualities and the performance comparison with other approaches show similar relative differences. Finally, we model the power consumption profile of the nodes after three devices characterized in Appendix A, namely "Soekris", "Alix" and "Linksys" whose parameters are summarized in Table 5.4.

5.4.1 Homogeneous Stations

The first scenario we study is a DOS network with homogeneous devices, that is, all the stations have the same power profiles. In order to evaluate the configuration strategy proposed in Section 5.3, we set up three different scenarios with the three devices of Table 5.4. For each scenario, we vary the total number of stations and compare the overall energy efficiency of our configuration ("Green DOS") against a configuration that maximizes overall throughput ("DOS") and a configuration that maximizes overall throughput but does not use opportunistic access ("non-opp.") in which all thresholds are zero. The results, depicted in Fig. 5.8, show that our strategy succeed in increasing the energy efficiency of the system for all cases. For the case of the "Soekris", the improvement in energy efficiency ranges from 30% with respect to the throughput optimization ("DOS")

[18] $p_{1(2)}$ is the access probability used by the 5 stations that belong to group 1(2) out of a total 10 heterogeneous stations in the toy experiment of Section 5.2.

and low number of stations to 10% in case of large number of stations. The "Alix" device, the least demanding device energy-wise, shows an increased performance of 9% comparing to the throughput optimization in case of two stations and a smaller improvement of 2% for large number of stations. In all cases, our strategy performs much better than the non-opportunistic configuration, with an energy-efficiency bump that ranges from 110% to 56%.

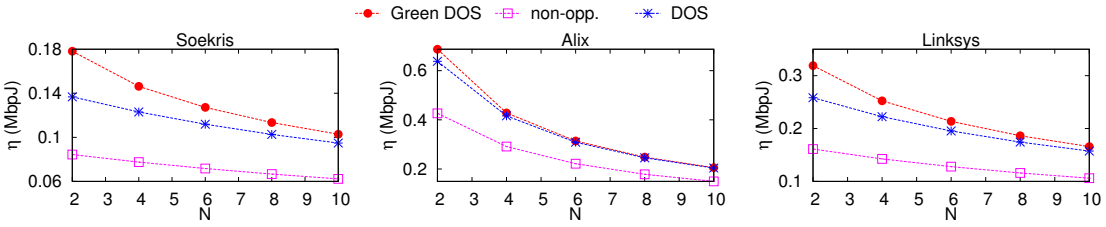


Figure 5.8: Energy efficiency with homogeneous stations.

5.4.2 Heterogeneous Stations

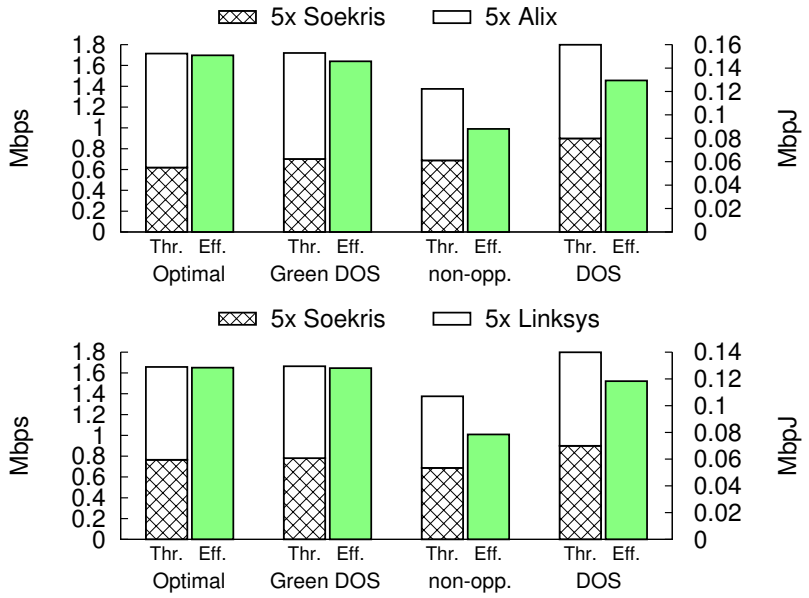


Figure 5.9: Energy efficiency with heterogeneous stations.

We set up now two heterogeneous scenarios with $N = 10$ stations, five of them modeled after the "Soekris" device; the remaining five stations are modeled after the "Alix" in the first scenario, and after the "Linksys" in the second scenario. We then compare, in Fig. 5.9, the throughput performance of each group of stations and the overall energy efficiency for the same configuration strategies as before, namely "DOS" and "non-opp.". To evaluate how close our heuristic strategy is from the EF-optimal configuration, we solved

offline the optimization problem described in (5.8) and include its outcome ("Optimal") in Fig. 5.9. The conclusion from these experiments is twofold. First, the heuristic strategy proposed in Section 5.3.2 follows very closely the performance of the optimal configuration providing an energy efficiency that approximates the offline optimization's by more than 95%. Second, the EF criterion represents a good tradeoff between maximizing overall energy efficiency and maintaining throughput fairness among the heterogeneous set of stations. The overall energy efficiency has been improved by approximately 15% with respect to the throughput optimization in both scenarios, and well above 50% comparing to the non-opportunistic configuration. We thus conclude that the strategy that we propose succeeds in balancing well fairness and an enhanced energy efficiency performance.

5.5 Summary

Energy-efficient operation of mobile devices is a key challenge for the design of future communication systems, which comprises the optimization of the energy consumption of wireless communications. In this chapter, we have performed an extensive experimental power consumption characterization of commodity wireless devices. While this work has been carried out through IEEE 802.11 (WiFi) interfaces, the results show compelling evidences that the behavior observed can be generalized to *any* commodity wireless device with "standard" software protocol stacks. Our study reveals a *new* power consumption cost associated to the software processing of every frame, apparently not accounted for by related literature, which we wrap into a convenient new energy model that we exploit in the design of an energy efficient strategy in DOS networks.

Then we have shown that heterogeneous scenarios (i.e., scenarios with stations featuring different energy consumption profiles) constitute a new research challenge, and we have identified the risk of extreme throughput unfairness if the optimization aims at maximizing only the overall efficiency. We thus have proposed a criterion to address this tradeoff between energy efficiency and throughput fairness, namely the EF criterion, and applied it in combination with optimal stopping theory in order to find and energy efficient strategy configuration for DOS-based networks with both homogeneous and heterogeneous devices. The criterion is not limited to this sort of networks but could be applied to any scenario and wireless technology with heterogeneous interfaces.

The proposed configuration have been validated through simulations (using a experimental-based parameterization of the node's energy profile), and has been shown to perform very similarly to the maximum achievable values derived from exhaustive searches on the configuration space.

Chapter 6

Conclusions and Future Work

Opportunistic scheduling mechanisms have been proposed to exploit, rather than mitigate, fading in wireless communications systems. These algorithms exploit knowledge about the channel quality distribution of all links in the network to preferentially allocate transmission opportunities on those links with best instantaneous channel conditions, thus improving overall performance. While the implementation of centralized schemes challenges practical use due to the quick radio quality variations associated to fading channels, a distributed approach, where users take their own decisions individually, faces the issue of coordination in such a dynamic environment.

Distributed Opportunistic Scheduling (DOS) has been proposed just recently to exploit fading links in distributed networks while skipping the need of explicit coordination among stations. The foundations of DOS were laid by Zheng et al. [7]; however, little attention has been paid to the actual design of adaptive mechanisms and other optimization objectives rather than total throughput performance. This thesis fills this gap by proposing three fundamental optimization criteria, and by designing the adaptive algorithms to support them.

In the first strategy we found the optimal setting of the DOS configuration parameters to provide a Proportional Fair (PF) allocation of individual throughputs in the network. We then proposed a distributed algorithm based in control theory, named ADOS, that adapts to the changing conditions of any wireless scenario, e.g., nodes entering and leaving or moving stations, and also works well in non-saturated scenarios, i.e,. with stations that does not always have data ready to transmit (indeed a common assumption in previous works). The design of this algorithm has been carefully addressed to provide a good tradeoff between stability and speed of convergence. Simulation results proved the effective behavior of ADOS in an extensive set of scenarios.

A problem inherent to the distributed nature of DOS mechanisms, however, is that the presence of rational users (non-cooperative) compromises the common welfare of the

system if they selfishly configure their DOS parameters to improve their individual share of resources, even advocating the network to performance collapse. Our game-theoretic analysis of DOS motivates the design of a penalizing mechanism to tackle misbehaving configurations. To this aim we proposed DOC, a distributed algorithm that naturally reacts upon these selfish behaviors by using a more aggressive configuration (than the optimal) in order to reduce the efficiency of the system to a point that removes any benefit from following other strategies (and thus the incentive to misbehave). A key feature of our punishing scheme is to carefully adjust the reaction against misbehaving strategies, as a miscalibrated maneuver could end up in an endless loop of reciprocal reactions which will settle the system in a suboptimal, or even unstable, point of operation. With DOC, we have used techniques from control theory and game theory to address this challenge and design an adaptive algorithm that (*i*) if all stations follow DOC, the system converges to the optimal PF allocation (and adapts to changing conditions such as mobility or joining/leaving stations), and (*ii*) one or more stations cannot benefit from following a different strategy, static or dynamic, individual or cooperative.

The aforementioned two algorithms target throughput performance and fairness as an optimization criteria. The third (an last) strategy proposed in this dissertation takes energy considerations into the objective performance figure as well. The heterogeneous nature of wireless equipment makes this goal very challenging as a naïve strategy aimed at *just* maximizing overall energy efficiency can penalize a subset of stations (the "power-hungry" ones, for example). We thus proposed a criterion, namely the EF criterion, to objectively balance an overall energy efficient optimization and throughput fairness. We then combined a novel power consumption model unveiled in this thesis and optimal stopping theory to devise an EF-optimal strategy for DOS networks comprised of both homogeneous and heterogeneous devices.

Even though our experimental study unveiled a model that maps very accurately the power consumption behavior of real-life wireless devices, the performance gains given by the opportunistic fading exploitation in the mechanisms devised in this thesis are yet to be validated empirically. Thus, as a future direction of this line of work, we study the possibility of experimentally testing DOS in real-life conditions by implementing DOS in *WARP* devices [115]. The WARP project provides an extensible programmable wireless platform for the prototyping of PHY protocols. Another line to explore towards this objective is the framework released by the EU project FLAVIA [116] that provides a flexible software-defined architecture very convenient for the prototyping of novel PHY/MAC designs like DOS. To this aim, a challenge that we would have to face is the design of a controlled environment for the deployment of our experiments to create on-demand fading environments that serve as a validation framework for the mechanisms under study in this thesis.

References

[1] C. Systems, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2010-2015," February 2011, White Paper. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.pdf

[2] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[3] S. Pettersson, "Radio resource management for wireless indoor communication systems: performance and implementation aspects," Ph.D. dissertation, KTH, Signals, Sensors and Systems, 2004.

[4] M. Cao, V. Raghunathan, and P. Kumar, "Cross-layer exploitation of mac layer diversity in wireless networks," in *Proc. of IEEE International Conference on Network Protocols (ICNP)*, Santa Barbara, CA, November 2006.

[5] P. Viswanath, D. N. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Trans. on Information Theory*, vol. 48, no. 6, pp. 1277–1294, June 2002.

[6] M. Andrews *et al.*, "Providing quality of service over a shared wireless link," *IEEE Communications Magazine*, vol. 39, no. 2, February 2001.

[7] D. Zheng, W. Ge, and J. Zhang, "Distributed opportunistic scheduling for ad hoc networks with random access: an optimal stopping approach," *IEEE Trans. on Information Theory*, vol. 55, no. 1, January 2009.

[8] P. Thejaswi *et al.*, "Distributed Opportunistic Scheduling With Two-Level Probing," *IEEE/ACM Trans. on Networking*, vol. 18, no. 5, October 2010.

[9] D. Zheng, *et al.*, "Distributed opportunistic scheduling for ad hoc communications with imperfect channel information," *IEEE Trans. on Wireless Communications*, vol. 7, no. 12, December 2008.

[10] S. Tan, D. Z. J. Zhang, and J. R. Zeidler, "Distributed Opportunistic Scheduling for Ad-Hoc Communications Under Delay Constraints," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, San Diego, CA, March 2010.

[11] T. S. Ferguson. Optimal Stopping and Applications. [Online]. Available: http://www.math.ucla.edu/~tom/Stopping/Contents.html

[12] H. Chen and J. Baras, "Distributed Opportunistic Scheduling for Wireless Ad-Hoc Networks with Block-Fading Model," *IEEE Journal on Selected Areas in Communications*, November 2013.

[13] H. Kim and G. de Veciana, "Losing opportunism: evaluating service integration in an opportunistic wireless system," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2007, pp. 982–990.

[14] M. J. Neely, "Opportunistic scheduling with worst case delay guarantees in single and multihop networks," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2011, pp. 1728–1736.

[15] R. K. Beidokhti, M. H. Y. Moghaddam, and J. Chitizadeh, "Adaptive QoS scheduling in wireless cellular networks," *Wireless Networks*, vol. 17, no. 3, pp. 701–716, 2011.

[16] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. on Telecommunications*, vol. 8, pp. 33–37, 1997.

[17] A. L. Toledo and X. Wang, "Robust detection of selfish misbehavior in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 6, pp. 1124–1134, August 2007.

[18] M. Raya, I. Aad, J.-P. Hubaux, and A. E. Fawal, "DOMINO: detecting MAC layer greedy behavior in IEEE 802.11 hotspots," *IEEE Trans. on Mobile Computing*, vol. 5, no. 12, pp. 1691–1705, December 2006.

[19] P. Serrano, A. Banchs, V. Targon, and J. F. Kukielka, "Detecting Selfish Configurations in 802.11 WLANs," *IEEE Communications Letters*, vol. 14, no. 2, pp. 142–144, February 2010.

[20] I. Tinnirello, L. Giarre, and G. Neglia, "MAC Design for WiFi Infrastructure Networks: A Game-Theoretic Approach," *IEEE Trans. on Wireless Communications*, vol. 10, no. 8, pp. 2510–2522, August 2011.

[21] M. Cagalj, S. Ganeriwal, I. Aad, and J.-P. Hubaux, "On selfish behavior in CSMA/CA networks," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, Miami, Florida, March 2005.

[22] J. Konorski, "A game-theoretic study of CSMA/CA under a backoff attack," *IEEE/ACM Trans. on Networking*, vol. 14, no. 6, pp. 1167–1178, 2006.

[23] A. Banchs, J. Ortin, A. Garcia-Saavedra, D. J. Leith, and P. Serrano, "Thwarting Selfish Behavior in 802.11e WLANs," *Submitted*.

[24] J.-M. Tarascon, "Key challenges in future Li-battery research," *Phil. Trans. of Royal Society A*, vol. 368, no. 1923, pp. 3227–3241, July 2010.

[25] G. Y. Li *et al.*, "Energy-efficient wireless communications: tutorial, survey, and open issues," *IEEE Wireless Communications*, vol. 18, no. 6, pp. 28–35, Dec. 2011.

[26] S.-L. Tsao and C.-H. Huang, "A survey of energy efficient MAC protocols for IEEE 802.11 WLAN," *Computer Communications*, vol. 34, no. 1, pp. 54–67, Jan. 2011.

[27] *IEEE 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std. 802.11, 2007.

[28] E. Rozner, V. Navda, R. Ramjee, and S. Rayanchu, "NAPman: network-assisted power management for WiFi devices," in *Proc. of ACM MobiSys.* ACM, 2010, pp. 91–106. [Online]. Available: http://doi.acm.org/10.1145/1814433.1814445

[29] D. Camps-Mur, A. Garcia-Saavedra, and P. Serrano, "Device to device communications with WiFi Direct: overview and experimentation," *IEEE Wireless Communications Magazine*, Accepted.

[30] X. He and F. Li, "Throughput and energy efficiency comparison of one-hop, two-hop, virtual relay and cooperative retransmission schemes," in *Proc. of European Wireless (EW)*, Apr. 2010, pp. 580–587.

[31] A. Garcia-Saavedra, B. Rengarajan, P. Serrano, D. Camps-Mur, and X. Pérez-Costa, "SOLOR: Self-Optimizing WLANs with Legacy-Friendly Opportunistic Relays," *Submitted.*

[32] A. P. Jardosh, K. Papagiannaki, E. M. Belding, K. C. Almeroth, G. Iannaccone, and B. Vinnakota, "Green WLANs: On-Demand WLAN Infrastructures," *Mobile Networks and Applications*, vol. 14, no. 6, pp. 798–814, Dec. 2009.

[33] D. Qiao, S. Choi, and K. Shin, "Interference analysis and transmit power control in IEEE 802.11a/h wireless LANs," *IEEE/ACM Trans. on Networking*, vol. 15, no. 5, pp. 1007–1020, Oct. 2007.

[34] J.-C. Chen and K.-W. Cheng, "EDCA/CA: Enhancement of IEEE 802.11e EDCA by contention adaption for energy efficiency," *IEEE Trans. on Wireless Communications*, vol. 7, no. 8, pp. 2866–2870, Aug. 2008.

[35] P. Serrano, A. Garcia-Saavedra, M. Hollick, and A. Banchs, "On the energy efficiency of IEEE 802.11 WLANs," in *Proc. of European Wireless (EW)*, 2010, pp. 932–939.

[36] A. Garcia-Saavedra, A. Banchs, P. Serrano, and J. Widmer, "Distributed Opportunistic Scheduling: A Control Theoretic Approach," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, Orlando, FL, March 2012.

[37] A. Garcia-Saavedra, A. Banchs, P. Serrano, and J. Widmer, "Distributed Opportunistic Scheduling: A Control Theoretic Approach," *Submitted.*

[38] A. Banchs, A. Garcia-Saavedra, P. Serrano, and J. Widmer, "A Game-Theoretic Approach to Distributed Opportunistic Scheduling," *IEEE/ACM Trans. on Networking*, Accepted.

[39] A. Garcia-Saavedra, P. Serrano, A. Banchs, and M. Hollick, "Energy-efficient fair channel access for IEEE 802.11 WLANs," in *Proc. of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, June 2011.

[40] A. Garcia-Saavedra, P. Serrano, A. Banchs, and M. Hollick, "Balancing energy efficiency and throughput fairness in IEEE 802.11 WLANs," *Pervasive and Mobile Computing*, vol. 8, no. 5, pp. 631–645, 2012.

[41] A. Garcia-Saavedra, P. Serrano, A. Banchs, and G. Bianchi, "Energy consumption anatomy of 802.11 devices and its implication on modeling and design," in *Proc. of ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, Dec. 2012, pp. 169–180.

[42] P. Serrano, A. Garcia-Saavedra *et al.*, "Per-frame Energy Consumption in 802.11 Devices and its Implication on Modeling and Design," *Submitted*.

[43] A. Garcia-Saavedra, P. Serrano, and A. Banchs, "An Energy-efficient Strategy using Distributed Opportunistic Scheduling," *Submitted*.

[44] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," *Proc. of ACM SIGCOMM*, vol. 34, no. 4, pp. 121–132, 2004.

[45] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge university press, 2005.

[46] R. Knopp and P. A. Humblet, "Information capacity and power control in single-cell multiuser communications," in *Proc. of IEEE International Conference on Communications (ICC)*, 1995, pp. 331–335.

[47] A. Jalali, R. Padovani, and R. Pankaj, "Data throughput of cdma-hdr a high efficiency-high data rate personal communication wireless system," in *Proc. of IEEE Vehicular Technology Conference (VTC)*, vol. 3. IEEE, 2000, pp. 1854–1858.

[48] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushyana, and S. Viterbi, "CDMA/HDR: a bandwidth efficient high speed wireless data service for nomadic users," *IEEE Communications Magazine*, vol. 38, no. 7, pp. 70–77, 2000.

[49] X. Liu, E. K. P. Chong, and N. B. Shroff, "Opportunistic transmission scheduling with resource-sharing constraints in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 10, pp. 2053–2064, 2001.

[50] R. Agrawal, A. Bedekar, R. J. La, R. S. Pazhyannur, and V. Subramanian, "Class and channel condition-based scheduler for EDGE/GPRS," in *Proc. of SPIE*, vol. 4531, 2001, pp. 59–69.

[51] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, and R. Vijayakumar, "Providing quality of service over a shared wireless link," *IEEE Communications Magazine*, vol. 39, no. 2, pp. 150–154, 2001.

[52] P. Viswanath, D. N. C. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Trans. on Information Theory*, vol. 48, no. 6, pp. 1277–1294, 2002.

[53] X. Qin and R. Berry, "Exploiting multiuser diversity for medium access control in wireless networks," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2003, pp. 1084–1094.

[54] S. Shakkottai, R. Srikant, and A. Stolyar, "Pathwise optimality and state space collapse for the exponential rule," in *Proc. of IEEE International Symposium on Information Theory (ISIT)*, 2002, p. 379.

[55] X. Liu, E. K. Chong, and N. B. Shroff, "A framework for opportunistic scheduling in wireless networks," *Computer Networks*, vol. 41, no. 4, pp. 451–474, 2003.

[56] N. Sharma and L. H. Ozarow, "A study of opportunism for multiple-antenna systems," *IEEE Trans. on Information Theory*, vol. 51, no. 5, pp. 1804–1814, 2005.

[57] A. Asadi and V. Mancuso, "A Survey on Opportunistic Scheduling in Wireless Communications," *IEEE Communications Surveys Tutorials*, 2013.

[58] M. Andrews and L. Zhang, "Scheduling algorithms for multi-carrier wireless data systems," in *Proc. of ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2007, pp. 3–14.

[59] S. Liu, L. Ying, and R. Srikant, "Scheduling in multichannel wireless networks with flow-level dynamics," in *Proc. of ACM Special Interest Group for the computer systems performance evaluation community (SIGMETRICS).*, vol. 38, no. 1, 2010, pp. 191–202.

[60] H. Al-Zubaidy, I. Lambadaris, and J. Talim, "Optimal scheduling in high-speed downlink packet access networks," *ACM Trans. on Modeling and Computer Simulation*, vol. 21, no. 1, p. 3, 2010.

[61] K. Khalil, M. Karaca, O. Ercetin, and E. Ekici, "Optimal scheduling in cooperate-to-join cognitive radio networks," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2011, pp. 3002–3010.

[62] C. Luo, F. R. Yu, H. Ji, and V. C. Leung, "Optimal channel access for tcp performance improvement in cognitive radio networks," *Wireless Networks*, vol. 17, no. 2, pp. 479–492, 2011.

[63] S. Liu, L. Ying, and R. Srikant, "Throughput-optimal opportunistic scheduling in the presence of flow-level dynamics," *IEEE/ACM Trans. on Networking*, vol. 19, no. 4, pp. 1057–1070, 2011.

[64] W. Ouyang, S. Murugesan, A. Eryilmaz, and N. B. Shroff, "Exploiting channel memory for joint estimation and scheduling in downlink networks," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2011, pp. 3056–3064.

[65] P. Jacko, "Value of information in optimal flow-level scheduling of users with Markovian time-varying channels," *Performance Evaluation*, vol. 68, no. 11, pp. 1022–1036, 2011.

[66] M. Ouyang and L. Ying, "On optimal feedback allocation in multichannel wireless downlinks," in *Proc. of ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, 2010, pp. 241–250.

[67] B. Sadiq, S. J. Baek, and G. De Veciana, "Delay-optimal opportunistic scheduling and approximations: The log rule," *IEEE/ACM Trans. on Networking*, vol. 19, no. 2, pp. 405–418, 2011.

[68] K. W. Choi, W. S. Jeon, and D. G. Jeong, "Resource allocation in OFDMA wireless communications systems supporting multimedia services," *IEEE/ACM Trans. on Networking*, vol. 17, no. 3, pp. 926–935, 2009.

[69] Z. Zhang, Y. He, and E. K. Chong, "Opportunistic downlink scheduling for multiuser OFDM systems," in *Proc. of IEEE Wireless Communications and Networking Conference*, vol. 2, 2005, pp. 1206–1212.

[70] J. Yang, Z. Yifan, W. Ying, and Z. Ping, "Average rate updating mechanism in proportional fair scheduler for HDR," in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, vol. 6, 2004, pp. 3464–3466.

[71] J.-T. Tsai, "State-dependent proportional fair scheduling algorithms for wireless forward link data services," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2008, pp. 2414–2422.

[72] Z. Shen, J. G. Andrews, and B. L. Evans, "Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints," *IEEE Trans. on Wireless Communications*, vol. 4, no. 6, pp. 2726–2737, 2005.

[73] J.-A. Kwon, B.-G. Kim, and J.-W. Lee, "A unified framework for opportunistic fair scheduling in wireless networks: a dual approach," *Wireless Networks*, vol. 16, no. 7, pp. 1975–1986, 2010.

[74] D. Fudenberg and J. Tirole, *Game Theory*. MIT Press, 1991.

[75] "Power Consumption and Energy Efficiency Comparisons of WLAN Products," White paper, Atheros Communications, Apr. 2004.

[76] "Data Transfer over Wireless LAN Power Consumption Analysis," White paper, Intel Corp., Feb. 2009.

[77] E. Lochin, A. Fladenmuller, J. yves Moulin, S. Fdida, and A. Manet, "Energy consumption models for ad-hoc mobile terminals," in *Proc. of Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, June 2003.

[78] A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone," in *Proc. of USENIX ATC*, June 2010, pp. 21–21.

[79] A. Rice and S. Hay, "Measuring mobile phone energy consumption for 802.11 wireless networking," *Pervasive and Mobile Computing*, vol. 6, no. 6, pp. 593–606, Dec. 2010.

[80] A. Pathak, Y. C. Hu, and M. Zhang, "Bootstrapping energy debugging on smartphones: a first look at energy bugs in mobile devices," in *Proc. of ACM Hot Topics in Networks (HotNets)*, Nov. 2011.

[81] L. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, Apr. 2001, pp. 1548–1557.

[82] J.-P. Ebert, B. Burns, and A. Wolisz, "A trace-based approach for determining the energy consumption of a WLAN network interface," in *Proc. of European Wireless (EW)*, Feb. 2002, pp. 230–236.

[83] E. Rantala, A. Karppanen, S. Granlund, and P. Sarolahti, "Modeling energy efficiency in wireless internet communication," in *Proc. of ACM SOSP Workshop on Networking, Systems, and Applications on Mobile Handhelds (MobiHeld)*, Aug. 2009, pp. 67–68.

[84] R. Bruno, M. Conti, and E. Gregori, "Optimization of efficiency and energy consumption in p-persistent CSMA-based wireless LANs," *IEEE Trans. on Mobile Computing*, vol. 1, no. 1, pp. 10–31, Mar. 2002.

[85] V. Baiamonte and C.-F. Chiasserini, "Saving energy during channel contention in 802.11 WLANs," *Mobile Networks and Applications*, vol. 11, no. 2, pp. 287–296, Apr. 2006.

[86] M. Ergen and P. Varaiya, "Decomposition of energy consumption in IEEE 802.11," in *Proc. of IEEE International Conference on Communications (ICC)*, June 2007, pp. 403–408.

[87] X. Wang, J. Yin, and D. P. Agrawal, "Analysis and optimization of the energy efficiency in the 802.11 DCF," *Mobile Networks and Applications*, vol. 11, no. 2, pp. 279–286, 2006.

[88] E.-S. Jung and N. H. Vaidya, "An Energy Efficient MAC Protocol for Wireless LANs," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, June 2002, pp. 1756–1764.

[89] M. Carvalho, C. Margi, K. Obraczka, and J. Garcia-Luna-Aceves, "Modeling energy consumption in single-hop IEEE 802.11 ad hoc networks," in *Proc. of International Conference on Computer Communications and Networks (ICCCN)*, Oct. 2004, pp. 367–372.

[90] D. Qiao, S. Choi, A. Jain, and K. G. Shin, "MiSer: an optimal low-energy transmission strategy for IEEE 802.11a/h," in *Proc. of ACM Mobicom*, Sept. 2003, pp. 161–175.

[91] H. Wu, S. Nabar, and R. Poovendran, "An energy framework for the network simulator 3 (NS-3)," in *Proc. of SIMUTools*, Mar. 2011, pp. 222–230.

[92] S. Chiaravalloti, F. Idzikowski, and L. Budzisz, "Power consumption of WLAN network elements," Tech. Univ. Berlin, Tech. Rep. TKN-11-002, Aug. 2011.

[93] G. Holland, N. Vaidya, and P. Bahl, "A Rate-adaptive MAC Protocol for multi-hop wireless networks," in *Proc. of ACM International Conference on Mobile Computing and Networking (MobiCom)*, Rome, Italy, July 2001.

[94] B. Sadhegi, V. Kanodia, A. Sabharwal, and E. Knightly, "Opportunistic Media Access for multirate ad hoc networks," in *Proc. of ACM International Conference on Mobile Computing and Networking (MobiCom)*, Atlanta, GA, September 2002.

[95] W. C. Jakes, *Microwave Mobile Communications.* New York: John Wiley & Sons Inc., 1975.

[96] P. Gupta, Y. Sankarasubramaniam, and A. Stolyar, "Random-Access Scheduling with Service Differentiation in Wireless Networks," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, Miami, FL, March 2005.

[97] B. Kristiansson and B. Lennartson, "Robust Tuning of PI and PID Controllers," *IEEE Control Systems Magazine*, vol. 26, no. 1, pp. 55–69, February 2006.

[98] A. K. Palit and D. Popovic, *Computational Intelligence in Time Series Forecasting: Theory and Engineering Applications.* Springer-Verlag New York, Inc., 2005.

[99] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*, 2nd ed. Addison-Wesley, 1990.

[100] G. Boggia, P. Camarda, L. A. Grieco, and S. Mascolo, "Feedback-based control for providing real-time services with the 802.11e MAC," *IEEE/ACM Trans. on Networking*, vol. 15, no. 2, April 2007.

[101] A. Banchs, P. Serrano, and L. Vollero, "Providing Service Guarantees in 802.11e EDCA WLANs with Legacy Stations," *IEEE Trans. on Mobile Computing*, vol. 9, no. 8, pp. 1057–1071, August 2010.

[102] C. V. Hollot, V. Misra, D. Towsley, and W. B. Gong, "A Control Theoretic Analysis of RED," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, Anchorage, Alaska, April 2001.

[103] K. Aström and B. Wittenmark, *Computer-controlled systems, theory and design*, 2nd ed. Prentice Hall International Editions, 1990.

[104] R. Jain, D. M. Chiu, and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems," DEC, Tech. Rep. TR-301, 1984.

[105] T. Glad and L. Ljung, *Control theory: multivariable and nonlinear methods.* Taylor & Francis, 2000.

[106] P. Patras, A. Banchs, P. Serrano, and A. Azcorra, "A Control Theoretic Approach to Distributed Optimal Configuration of 802.11 WLANs," *IEEE Trans. on Mobile Computing*, vol. 10, no. 6, June 2011.

[107] J. S. Milton and J. C. Arnold, *Introduction to probability and statistics, 4th ed.* McGraw-Hill Higher Education, 2003.

[108] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, Mar 2000.

[109] J. L. Lee *et al.*, "An experimental study on the capture effect in 802.11a networks," in *Proc. of ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*, Sept. 2007.

[110] Y. Chen, S. Zhang, S. Xu, and G. Li, "Fundamental trade-offs on green wireless networks," *IEEE Communications Magazine*, vol. 49, no. 6, pp. 30–37, 2011.

[111] H. Wang, N. Agoulmine, M. Ma, and Y. Jin, "Network lifetime optimization in wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 1127 –1137, september 2010.

[112] L. Massouli and J. Roberts, "Bandwidth sharing and admission control for elastic traffic," *Telecommunication Systems*, vol. 15, pp. 185–201, 2000.

[113] T. Nandagopal, T.-E. Kim, X. Gao, and V. Bharghavan, "Achieving MAC layer fairness in wireless packet networks," in *Proc. of ACM International Conference on Mobile Computing and Networking (MobiCom)*, New York, USA, 2000, pp. 87–98.

[114] A. Banchs, P. Serrano, and H. Oliver, "Proportional fair throughput allocation in multirate IEEE 802.11e wireless LANs," *Wireless Networks*, vol. 13, no. 5, Oct 2007.

[115] P. Murphy, A. Sabharwal, and B. Aazhang, "Design of WARP: A Flexible Wireless Open-Access Research Platform," in *Proc. of European Signal Processing Conference (EUSIPCO)*, 2006.

[116] FLAVIA Project: FLexible Architecture for Virtualizable future wireless Internet Access. [Online]. Available: http://www.ict-flavia.eu/

[117] S. Keranidis *et al.*, "Energy consumption monitoring of nitos testbed nics through the nitos emf framework," in *Proc. of ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization (WiNTECH)*, Sept. 2013.

[118] P. Bahl, R. Chandra, P. Lee, V. Misra, J. Padhye, D. Rubenstein, and Y. Yu, "Opportunistic use of client repeaters to improve performance of WLANs," *IEEE/ACM Trans. on Networking*, vol. 17, no. 4, pp. 1160–1171, Aug. 2009.

[119] C. Sun and C. Yang, "Is two-way relay more energy efficient?" in *Proc. of IEEE Global Communications Conference (GLOBECOM)*, Dec. 2011.

[120] M. Hosseini, D. Ahmed, S. Shirmohammadi, and N. Georganas, "A survey of application-layer multicast protocols," *IEEE Communications Surveys and Tutorials*, vol. 9, no. 3, pp. 58 –74, Sept. 2007.

[121] *IEEE 802.11: Amendment 2. MAC Enhancements for Robust Audio Video Streaming*, IEEE Std. 802.11aa, 2012.

[122] A. B. Sharma, L. Golubchik, R. Govindan, and M. J. Neely, "Dynamic data compression in multi-hop wireless networks," in *Proc. of ACM Special Interest Group for the computer systems performance evaluation community (SIGMETRICS).*, June 2009, pp. 145–156.

[123] S. J. Baek, G. de Veciana, and X. Su, "Minimizing energy consumption in large-scale sensor networks through distributed data compression and hierarchical aggregation," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1130–1140, Aug. 2004.

[124] A. Jain and M. Gruteser, "Benefits of packet aggregation in ad-hoc wireless network," Univ. of Colorado at Boulder, Tech. Rep., Aug. 2003.

[125] F. Qian *et al.*, "Characterizing radio resource allocation for 3G networks," in *Proc. of ACM Internet Measurement Conference (IMC)*, Nov. 2010, pp. 137–150.

[126] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: A measurement study and implications for network applications," in *Proc. of ACM Internet Measurement Conference (IMC)*, Nov. 2009.

[127] D. Bonfiglio, M. Mellia, M. Meo, N. Ritacca, and D. Rossi, "Tracking down Skype traffic," in *Proc. of IEEE International Conference on Computer Communications (INFOCOM)*, 2008, pp. 261–265.

Appendix A

Power Consumption in 802.11 Devices and its Implication on Modeling and Design

A.1 Energy measurement testbed

In our experimental analysis, baseline results are obtained using a Soekris net4826-48 device, equipped with an Atheros AR5414-based 802.11a/b/g Mini-PCI card, and configured to use the 802.11a PHY. The hardware comprises a 233MHz AMD Geode SC1100 CPU, 2 Mini-PCI sockets, 128 Mbyte SDRAM and 256 Mbyte compact flash circuits for data storage, extended with a 2 GB USB drive. The OS is a Gentoo 10.0 Linux (kernel 2.6.24), and the driver is MadWifi v0.9.4.

To rule out the possibility that our findings could be biased by the selected HW, OS/driver SW, or WLAN band/card/PHY, most of the thesis's experiments have been performed for other platforms (see Table 5.1). This set of platforms include many different types of devices, such as wireless routers, Access Points, smart phones, tablets and embedded devices.[1]

In all experiments, traffic was generated using `mgen`.[2] Additional devices in monitor mode were employed to track wireless channel activity and confirm it was caused only by our experiments, and that no packets were dropped at any layer of the protocol stack, events which would have biased our findings.

[1] In addition to the results that we report here for different platforms, the generality of our model is further confirmed by the measurements recently carried out independently in [117]. Building on our previous work of [41], these measurements show that an Atom-based node equipped with an AR9380 NIC also follows our energy consumption model.

[2] `http://cs.itd.nrl.navy.mil/work/mgen/`

Depending on the platform, power consumption was measured via two instruments. The **Monsoon FTA22D** power meter[3] supplies a stable voltage to the device and samples the power consumption at 5 KHz with high accuracy (±1%), thus providing very reliable fine-grained power measurements. However, this meter only supplies voltages of up to 5.4 V and could thus be employed only for the tablet Samsung Galaxy Note 10.1, the smartphone HTC Legend and the Raspberry Pi. Measurements for the Soekris, Alix and Linksys platforms, which require higher voltage, were carried out using a **PCE PA-6000** power meter,[4] which has a much coarser granularity than the Monsoon meter (namely about 3 samples/second). The PCE meter was powered with 6 AA batteries (AC supply via the wall socket would reduce accuracy), and we employed a Protek 3033B device to power the wireless device. Both the Monsoon and the PCE meters permit to perform measurements without *dismantling* the device, as required by some specialized equipments, which thus may restrict experimentation to, e.g., devices using card extenders.

A major practical challenge was to reduce the errors associated to the measurements of the PCE PA-6000 power meter, which natively provides an accuracy of $\Delta v = 0.1\,V$ for the voltage and $\Delta i = 0.01\,A$ for the current. Taking into account the well-known fact that the relative error for the product $p = v \cdot i$ is approximated by the sum of the relative errors for v and i, these inaccuracies yield a relative error in the measured power of above 5% for our typical baseline power measurements, undermining our ability to quantify small, but for our purposes meaningful, trends (e.g., power consumption variations for an increased frame size). The methodology that we followed to improve the accuracy of our measurements is to use, instead of a single device, K devices *running the same experiment* in parallel over different non-interfering wireless channels. With this, the relative error of the measurements is reduced by $1/K$. For 802.11g, we used the three non interfering channels, whereas we used $K = 4$ for 802.11a, thus improving accuracy to about 2% (see [41] for a more detailed explanation).

A.2 The cross-factor

To grasp a deeper insight on the reasons behind the cross-factor γ^{xg}, we have engineered tailored measurements devised to *quantify* how the energy toll splits across the frame processing chain along the protocol stack implementation (roughly) depicted in Fig. A.1. Specifically, we have run three sets of experiments, where we discard packets at a given level of the stack and measure the power consumed up to that level:

- **App.** - packets are generated by `mgen`, but are discarded before being delivered to the OS, i.e., at the mark (a) in Fig. A.1, by sending them to the "sink device"

[3] `http://www.msoon.com/LabEquipment/PowerMonitor/`

[4] `http://www.industrial-needs.com/technical-data/power-analyser-PCE-PA-6000.htm`

(`/dev/null`);

- **TCP/IP** - packets are discarded at the bottom of the TCP/IP stack (mark (*b*) in Fig. A.1), by deactivating the ARP lookup function, so that the device cannot retrieve the MAC destination from the ARP cache and therefore must drop the frame;
- **Driver** - packets are discarded after the MadWifi driver's processing (mark (*c*) in Fig. A.1), by commenting the `hardstart` command which performs the actual delivery of the frame to the NIC.

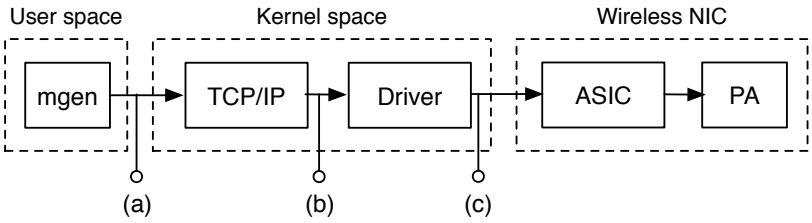


Figure A.1: Interfaces/modules crossed during transmission.

Representative measurements (energy per frame) are shown in Fig. A.2, along with the total energy consumption per properly transmitted frame ('Total') and the values predicted by applying (5.2) ('Model').

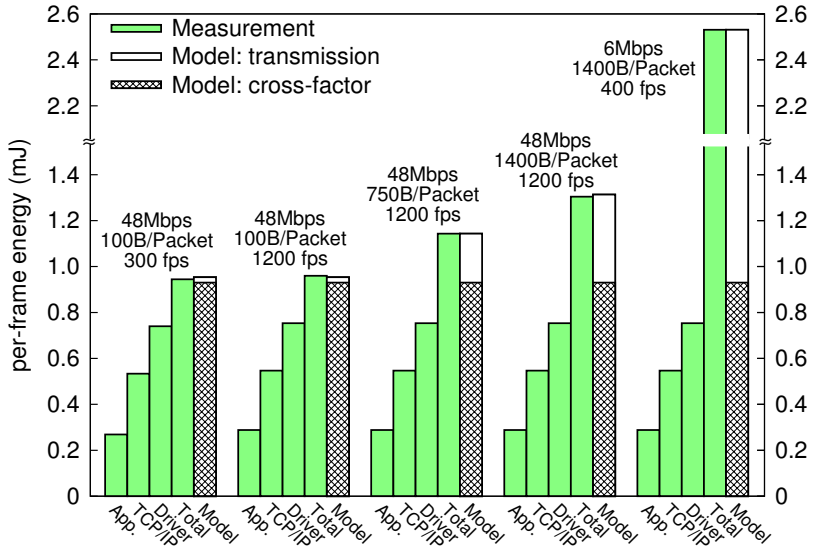


Figure A.2: Per-frame energy cost in transmission.

The figure clearly shows that the energy toll due to frame processing is practically *independent of* the frame generation rate *and* the frame size. Moreover, it shows that the

energy consumed while crossing the host device stack (i.e., up to the driver included) is *substantial*, around 0.75 mJ per frame, and may become the *major* energy cost in several scenarios (e.g., short packets and/or large MCS - in essence short airtime).

Finally, even if direct measurements cannot be attained below the driver level, Fig. A.2 shows that a further constant per-frame energy drain occurs at the driver-to-NIC interface level and/or below. Its quantification may be estimated by measuring the energy consumed with very short packets and large MCS, as wireless transmission cost is marginal in this case (very small airtime). Summarizing, for the Soekris device, the cross-factor coefficient amounts to about 0.93 mJ/frame. Such per-frame processing cost appears to roughly split as follows: 24% application; 33% TCP/IP stack, 21% driver, and 22% NIC.

A.3 Analysis of IEEE 802.11-specific transmission costs

A.3.1 Retransmissions

The above results seem to suggest that retransmissions at the MAC layer, e.g. caused by an unacknowledged transmission, should not be affected by the cross-factor toll. This can be verified by *provisionally assuming* that this is the case, i.e., modeling retransmission cost as purely due to the over the air transmission cost component, and then checking whether the resulting model matches experimental measurements. Hence, let P^{retx} be the power drained by retransmissions, and assume that

$$P^{retx} = R \cdot \pi^{tx} \alpha^{tx} = R \cdot \pi^{tx} \lambda^{g} T_L. \tag{A.1}$$

where R is the number of retransmissions, λ^g is the frame generation rate, and $T_L = T_{PLCP} + (H + L)/MCS$ is the time required to transmit a frame of size L using the modulation and coding scheme MCS, duly accounting for the Physical Layer Convergence Protocol preamble T_{PLCP}, and the MAC overhead H (MAC header plus FCS). Then, the *total* power consumed by packets retransmitted R times is readily obtained as the baseline component π^{id} plus:

$$P^{xg}(\lambda^g) + P^{tx} + P^{retx}, \tag{A.2}$$

where the first addendum is the per-frame processing toll (paid once), the second addendum is the power consumed by the very first transmission, and the last addendum is the extra retransmission cost as per (A.1). Fig. A.3 compares the modeling prediction of (A.2) with the power (additional to the baseline component π^{id}) consumed by a device configured to send 1400 B frames generated at a rate of 80 fps to fake addresses (to prevent the reception of ACKs). The number of allotted retransmissions R (configured via the `ah_setupTxDesc` driver's descriptor) was varied from 0 to 5, and, for simplicity (i.e.,

to avoid the need to non trivially configure the driver so as to prevent MCS downgrade in front of persistent losses), frames were transmitted using the 6 Mbps basic MCS. As shown in Fig. A.3, theoretical results tightly match the experimental measurements, thus confirming that the cross-factor has (if any) a negligible impact on retransmissions.

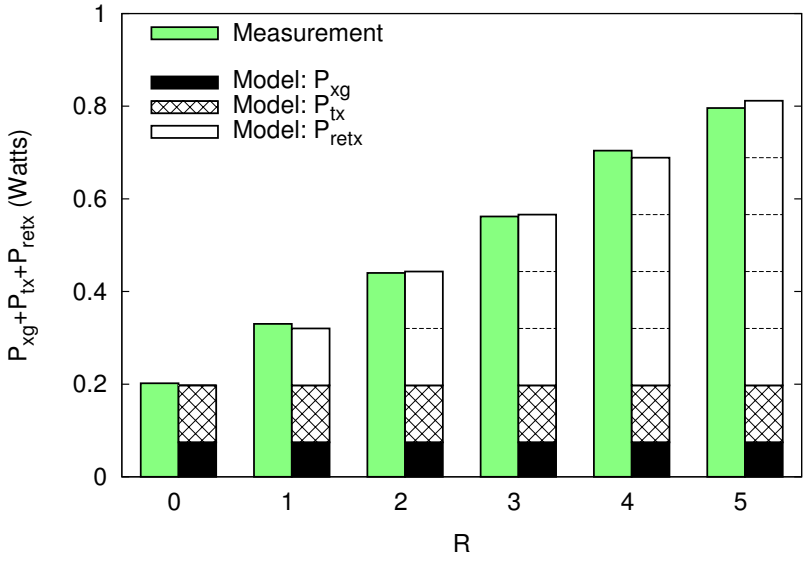


Figure A.3: Impact of retransmissions on power consumption.

A.3.2 Characterization of ACKs and other control frames

To complete our analysis, it remains to characterize the additional power consumed for sending/receiving acknowledgments (the previous experiments have considered unacknowledged operation) and other control frames.

Since ACK frames, like retransmissions, do not have to cross the stack but are internally generated by the NIC, we make the hypothesis that their power consumption can be characterized by just the cost of the relevant ACK transmission or reception. Under such hypothesis, the power consumed for replying with ACKs to received frames (arriving at rate λ^r) is trivially given by

$$P^{tx,Ack} = \pi^{tx} \lambda^r T_{Ack}, \tag{A.3}$$

where $T_{Ack} = T_{PLCP} + ACK/MCS_C$ is the time required to transmit an ACK frame, i.e., a PLCP preamble plus the 14B ACK frame transmitted at the modulation and coding scheme MCS_C configured for control traffic. Similarly, the power consumed to receive an ACK is readily computed as

$$P^{rx,Ack} = \pi^{rx} \lambda^g T_{Ack}. \tag{A.4}$$

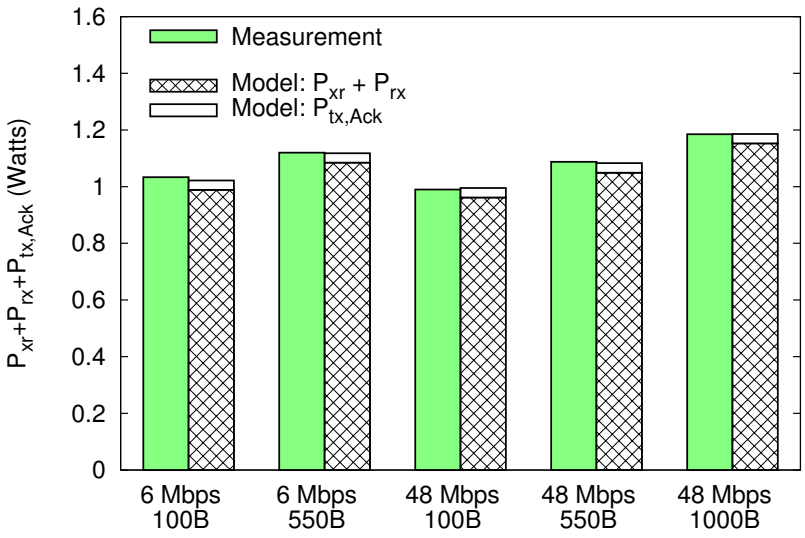


Figure A.4: Impact of sending ACKs on the receiver.

Fig. A.4 shows the experimental validation for the ACK transmission case. Such experimental results, obtained with $\lambda^r = 1000$ fps, confirm that the measurements match the results predicted by the model, which includes the energy consumed by the reception of frames ($P^{xr} + P^{rx}$) and the transmission of the ACKs ($P^{tx,Ack}$). Similar findings hold for RTS and CTS frames (not shown here for space reasons).

A.4 Parametrization of the model

In order to characterize the power consumption of an wireless device, we need to parametrize the 5 constant parameters in (5.5) for the device. We have obtained of these 5 parameters for all the devices under study: Soekris (with Linux and OpenBSD), Alix, Linksys, HTC Legend, Samsung Galaxy Note and Raspberry Pi. The numerical values are given in Table A.1 for the different MCS and `txpower` configurations, as well as different CPU frequencies for those devices that support CPU scaling. To obtain these values, we have applied the expressions for the simple linear regression and the standard asymptotic error [107]. Some observations revealed by our parametrization are:

- Most cards show moderate differences of π^{tx} for different MCS, except for the Soekris (the only device with Atheros), whose power consumption increases substantially for large MCS. In contrast, differences are significant for all devices in reception mode.
- As expected, receiving (π^{rx}) consumes substantially less power than transmitting

Table A.1: Parametrization of the energy model for all the devices under study.

Soekris	MCS			6 Mbps	12 Mbps	24 Mbps	48 Mbps
	π^{rx} (W)			$0.16 \pm 8\%$	$0.27 \pm 5.6\%$	$0.6 \pm 11\%$	$1.14 \pm 3.5\%$
	π^{tx} (W)		6 dBm	$0.52 \pm 3.1\%$	$0.55 \pm 4.6\%$	$0.81 \pm 5.3\%$	$1.2 \pm 1.6\%$
			9 dBm	$0.57 \pm 2.1\%$	$0.59 \pm 1.8\%$	$0.88 \pm 2.3\%$	$1.24 \pm 2.7\%$
			12 dBm	$0.70 \pm 1.7\%$	$0.73 \pm 2.2\%$	$1.02 \pm 2.8\%$	$1.37 \pm 3.1\%$
			15 dBm	$0.86 \pm 2.2\%$	$0.89 \pm 2.3\%$	$1.17 \pm 2.5\%$	$1.58 \pm 3.3\%$
	π^{id} (W)	Linux	$3.56 \pm 0.6\%$	γ^{xg} (mJ)	$0.93 \pm 1.2\%$	γ^{xr} (mJ)	$0.93 \pm 2.2\%$
		OpenBSD	$3.48 \pm 0.8\%$		$1.27 \pm 1.9\%$		$1.26 \pm 2.0\%$
Linksys	MCS			6 Mbps	12 Mbps	24 Mbps	48 Mbps
	π^{rx} (W)			$0.19 \pm 5.3\%$	$0.29 \pm 3.4\%$	$0.53 \pm 2.3\%$	$0.74 \pm 4.4\%$
	π^{tx} (W)		6 dBm	$0.70 \pm 1.1\%$	$0.72 \pm 2.2\%$	$0.75 \pm 2.0\%$	$0.81 \pm 3.7\%$
			9 dBm	$0.77 \pm 1.4\%$	$0.81 \pm 2.6\%$	$0.84 \pm 2.3\%$	$0.88 \pm 3.4\%$
			12 dBm	$0.84 \pm 1.2\%$	$0.85 \pm 1.5\%$	$0.92 \pm 2.4\%$	$0.99 \pm 4.0\%$
			15 dBm	$0.97 \pm 0.9\%$	$1.0 \pm 1.5\%$	$1.04 \pm 2.1\%$	$1.08 \pm 3.7\%$
	π^{id} (W)		$2.73 \pm 0.4\%$	γ^{xg} (mJ)	$0.46 \pm 3.3\%$	γ^{xr} (mJ)	$0.43 \pm 4.2\%$
Alix	MCS			6 Mbps	12 Mbps	24 Mbps	48 Mbps
	π^{rx} (W)			$0.24 \pm 4.2\%$	$0.27 \pm 3.7\%$	$0.31 \pm 6.4\%$	$0.44 \pm 6.8\%$
	π^{tx} (W)		6 dBm	$0.27 \pm 7.4\%$	$0.33 \pm 9.1\%$	$0.35 \pm 11.4\%$	$0.38 \pm 5.2\%$
			9 dBm	$0.30 \pm 6.7\%$	$0.35 \pm 8.6\%$	$0.36 \pm 11.1\%$	$0.39 \pm 5.3\%$
			12 dBm	$0.35 \pm 5.7\%$	$0.38 \pm 7.9\%$	$0.39 \pm 7.7\%$	$0.43 \pm 7.0\%$
			15 dBm	$0.4 \pm 7.5\%$	$0.44 \pm 6.8\%$	$0.45 \pm 8.9\%$	$0.46 \pm 8.7\%$
	π^{id} (W)		$3.68 \pm 0.5\%$	γ^{xg} (mJ)	$0.11 \pm 7.6\%$	γ^{xr} (mJ)	$0.09 \pm 8.5\%$
HTC Legend	MCS			6 Mbps	12 Mbps	24 Mbps	48 Mbps
	π^{rx} (mW)			$17.08 \pm 9.0\%$	$44.51 \pm 4.8\%$	$90.52 \pm 5.22\%$	$135.79 \pm 16.9\%$
	π^{tx} (mW)		6 dBm	$296.93 \pm 1.2\%$	$325.84 \pm 2.4\%$	$376.72 \pm 1.0\%$	$393.82 \pm 6.9\%$
			9 dBm	$315.69 \pm 0.8\%$	$347.50 \pm 2.5\%$	$385.69 \pm 2.6\%$	$402.50 \pm 5.0\%$
			12 dBm	$346.17 \pm 0.4\%$	$353.76 \pm 1.8\%$	$398.33 \pm 1.4\%$	$418.74 \pm 4.6\%$
			15 dBm	$381.94 \pm 0.8\%$	$396.22 \pm 1.7\%$	$436.28 \pm 2.3\%$	$447.25 \pm 3.3\%$
	π^{id} (mW)	245MHz	$548.48 \pm 0.1\%$	γ^{xg} (mJ)	$0.0126 \pm 5.5\%$	γ^{xr} (mJ)	$0.005 \pm 17.0\%$
		480MHz	$700.97 \pm 0.2\%$		$0.0127 \pm 6.3\%$		$0.005 \pm 17.3\%$
		600MHz	$874.66 \pm 0.3\%$		$0.0131 \pm 2.7\%$		$0.006 \pm 14.1\%$
Samsung Galaxy Note 10.1	MCS			6 Mbps	12 Mbps	24 Mbps	48 Mbps
	π^{rx} (mW)		600MHz	$16.16 \pm 13.8\%$	$21.22 \pm 12.0\%$	$38.93 \pm 16.5\%$	$51.82 \pm 14.12\%$
			1GHz	$24.51 \pm 13.1\%$	$40.70 \pm 9.9\%$	$48.47 \pm 6.1\%$	$71.72 \pm 9.5\%$
			1.4GHz	$54.08 \pm 12.0\%$	$58.20 \pm 11.5\%$	$82.35 \pm 6.6\%$	$124.19 \pm 13.3\%$
	π^{tx} (mW)	6 dBm	600MHz	$594.88 \pm 2.8\%$	$614.11 \pm 1.6\%$	$620.51 \pm 5.4\%$	$628.77 \pm 2.1\%$
			1GHz	$617.1 \pm 1.7\%$	$618.52 \pm 2.9\%$	$632.70 \pm 2.6\%$	$661.64 \pm 2.8\%$
			1.4GHz	$656.36 \pm 1.0\%$	$689.69 \pm 2.2\%$	$744.87 \pm 1.8\%$	$781.77 \pm 4.3\%$
		9 dBm	600MHz	$604.13 \pm 2.6\%$	$627.13 \pm 0.5\%$	$627.03 \pm 1.9\%$	$636.15 \pm 2.0\%$
			1GHz	$620.07 \pm 1.8\%$	$629.74 \pm 2.7\%$	$661.58 \pm 2.3\%$	$697.64 \pm 3.2\%$
			1.4GHz	$673.71 \pm 5.4\%$	$706.67 \pm 3.4\%$	$755.65 \pm 3.9\%$	$786.48 \pm 2.6\%$
		12 dBm	600MHz	$621.27 \pm 0.9\%$	$628.09 \pm 1.6\%$	$632.16 \pm 1.6\%$	$648.33 \pm 2.3\%$
			1GHz	$626.31 \pm 1.4\%$	$639.88 \pm 1.6\%$	$675.01 \pm 1.8\%$	$718.10 \pm 1.4\%$
			1.4GHz	$689.62 \pm 3.9\%$	$711.45 \pm 1.1\%$	$761.48 \pm 1.9\%$	$791.99 \pm 3.8\%$
		15 dBm	600MHz	$630.07 \pm 2.5\%$	$630.47 \pm 2.8\%$	$644.00 \pm 1.5\%$	$670.41 \pm 2.4\%$
			1GHz	$645.42 \pm 4.5\%$	$652.61 \pm 1.6\%$	$689.77 \pm 2.4\%$	$746.77 \pm 3.1\%$
			1.4GHz	$699.27 \pm 7.4\%$	$717.18 \pm 2.8\%$	$768.96 \pm 2.4\%$	$809.10 \pm 6.3\%$
	π^{id} (mW)	600MHz	$581.21 \pm 0.07\%$	γ^{xg} (mJ)	$0.045 \pm 1.6\%$	γ^{xr} (mJ)	$0.048 \pm 16.8\%$
		1.0GHz	$885.85 \pm 0.1\%$		$0.056 \pm 0.8\%$		$0.059 \pm 5.2\%$
		1.4GHz	$1194.29 \pm 0.1\%$		$0.088 \pm 1.3\%$		$0.098 \pm 6.7\%$
Raspberry Pi	MCS			6 Mbps	12 Mbps	24 Mbps	48 Mbps
	π^{rx} (mW)			$5.1 \pm 35.3\%$	$6.5 \pm 23.6\%$	$31.6 \pm 23.7\%$	$63.4 \pm 8.06\%$
	π^{tx} (mW)		6 dBm	$593.6 \pm 0.45\%$	$583.3 \pm 0.82\%$	$565.2 \pm 0.96\%$	$599.7 \pm 0.84\%$
			9 dBm	$627.7 \pm 0.37\%$	$611.8 \pm 1.37\%$	$587.5 \pm 1.28\%$	$621.9 \pm 2.14\%$
			12 dBm	$687.8 \pm 0.70\%$	$674.6 \pm 0.84\%$	$653.82 \pm 1.17\%$	$693.7 \pm 1.56\%$
			14 dBm	$692.2 \pm 1.85\%$	$716.3 \pm 0.85\%$	$748.8 \pm 3.7\%$	$806.6 \pm 2.56\%$
	π^{id} (mW)		$2220.3 \pm 0.13\%$	γ^{xg} (mJ)	$0.126 \pm 3.34\%$	γ^{xr} (mJ)	$0.049 \pm 5.78\%$

(π^{tx}) in most cases. However, for some of the devices (Soekris, Linksys and Alix), receiving is almost as costly as transmitting, or even more, for large MCS.

- While the difference on the energy consumed by different devices is relatively small in transmission mode (π^{tx} is of the same order of magnitude for all devices), the differences in reception mode are much higher (differences in π^{rx} are larger than one order of magnitude).
- The CPU frequency has little impact on the power consumed by the NIC-specific parts: the values of π^{tx} and π^{rx} are the same for different CPU frequencies with the HTC Legend (only one value is given in the table) and they are also similar for the Samsung Galaxy Note.
- The cross-factor for sending and receiving (γ^{xg} and γ^{xr}) is of the same order of magnitude in all devices.

A.5 Model extension for TCP

The experiments performed so far, as well as the proposed model, have been restricted to UDP. Arguably, we expect quantitative differences in the cross-factor emerging with TCP traffic, due to the increased stack processing complexity. To assess these differences, we separately study TCP segments and TCP ACKs. For measuring the power consumed by a TCP data packet, we have modified the TCP stack as follows: (i) at the receiver side, we do not send TCP ACKs; and (ii) at the sender side, we have deactivated the TCP timers as well as the checks of congestion and receive window. With these modifications, the energy consumed by the sender is caused by the processing of TCP segments only, which are sent at the rate given by the application layer (since congestion control is not activated). By performing a similar experiment to the one of Fig. 5.1 with this modified TCP stack, we have measured the cross-factor associated to the processing of TCP segments, which is of $\gamma^{xg\_tcp} = 1.38mJ$ (all other energy components remain the same), indeed almost 50% larger than that previously measured for UDP.

To further evaluate the energy consumed by processing the TCP ACKs, we have repeated the same experiment as above with the unmodified TCP stack, with results in a cross-factor of $2.1mJ$. Since the main difference between the modified TCP stack and the unmodified one is the processing of TCP acknowledgements, by subtracting from this cross-factor the one measured above we obtain the energy toll for receiving TCP ACKs, i.e. $\gamma^{xr\_tcpack} = 2.1 - 1.38 = 0.72mJ$. Note that this result is in line with the one above, since as compared to TCP segments, TCP ACKs require less processing and do not involve the application layer (which, as shown in Fig. A.2, accounts for a significant portion of the cross-factor).

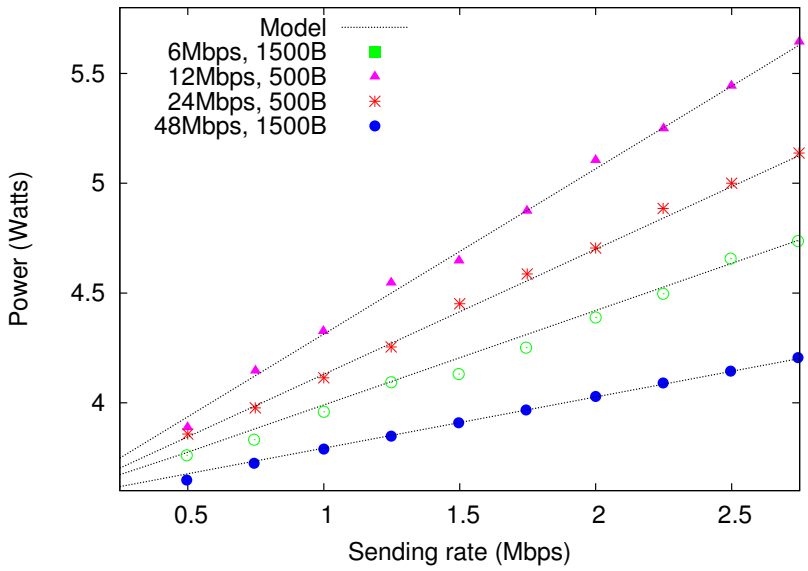


Figure A.5: Model validation for TCP.

Following the above findings, we can extend model (5.5) as follows in order to account for all traffic types:

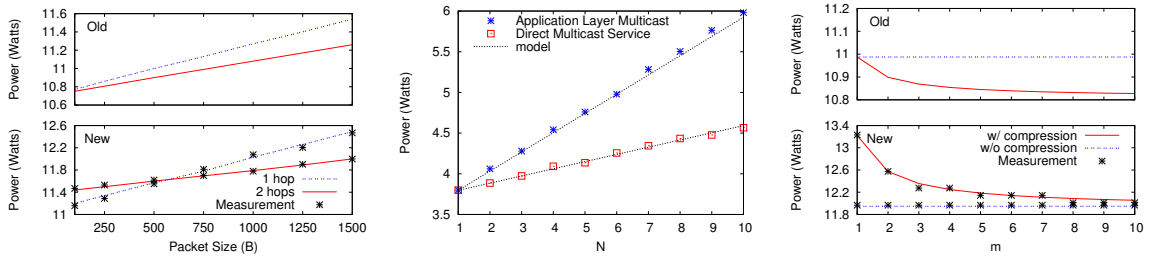
$$\mathbf{P} \quad = \quad \pi^{id} + \pi^{tx}\alpha^{tx} + \pi^{rx}\alpha^{rx} + \sum_{i \in \mathcal{K}} \gamma^i \lambda^i. \tag{A.5}$$

where γ^i and λ^i are the cross-factor and sending rate, respectively, and $\mathcal{K}$ is the set of different frame types, which includes transmitted and received UDP packets as well as TCP data packets and TCP ACKs, i.e., $\mathcal{K} = \{xg, xr, xg\_tcp, xr\_tcp, xg\_tcpack, xr\_tcpack\}$.

In order to validate the extended model, we have measured the energy consumed by a TCP session between two stations for different different configurations of the MCS, packet size and sending rate. The results on the energy consumed by the sender, depicted in Fig. A.5, show that the energy consumption predicted by the model closely matches our measurements, which confirms the accuracy of the proposed extension.

A.6 Implications on design

The new energy consumption insights gathered in this thesis may have significant implications on the design of energy-efficient mechanisms. On the one hand, existing schemes may need to be revisited so as to properly account for the impact of the cross-factor component. Indeed, according to traditional power consumption models (i.e., only baseline component plus a toll proportional to the airtime), mechanisms yielding shorter airtimes would *surely* bring about energy gains. With the cross factor, this *might not be*



(a) Packet relaying. (b) Multicasting. (c) Data compression in multi-hop network.

Figure A.6: Revisiting previous schemes under the new model.

anymore the case, when the power savings attained at the radio interface are paid with an increased frame handling and its associated (*non marginal*) power consumption. On the other hand, the gained knowledge that a frame crossing the stack brings about a *fixed* penalty unrelated to the frame size may be exploited to devise techniques to *avoid* or *reduce* such energy toll.

In the following, with no pretense of completeness, we present *quantitative* examples that show how our new insights may affect existing energy efficient mechanisms as well as inspire novel approaches. As previously, we first conduct experiments with the Soekris/Linux baseline platform, and then extend them to all other devices under study.

A.6.1 Reconsidering existing schemes

Packet relaying

Packet relaying in WLANs is commonly used to improve performance [118] and energy efficiency [30]. The rationale is that the use of a relay permits shorter transmission times, which compensate the impact of the extra number of hops, thus introducing a net gain. However, classical energy-efficiency analyses do not balance the airtime energy saving with the energy drain introduced by the additional frame processing, a penalty which may *fundamentally affect the relevant conclusions.*

To quantitatively support this claim, we deployed a two-hop scenario comprising three nodes (sender, relay and receiver), and compared the power consumption in two different configurations (taken from [118]): (*i*) traffic directly sent to the receiver (1-hop, at 6 Mbps), and (*ii*) relay node used (2-hops, both at 48 Mbps). Traffic is generated at a rate of $\lambda^g = 400$ fps with `txpower` $= 15$ dBm and different frame sizes L. Packet forwarding in the relay is performed at the routing layer. In both configurations, the relay node is always active.[5]

[5]In most of the analyses on energy efficiency of relaying, the relay does not use the "sleep mode" (see e.g. [30, 119]).

Three types of results are shown in Fig. A.6a: (*i*) experimental measurements, (*ii*) theoretical predictions using a traditional model that neglects the impact of crossing the protocol stack ('old'), and (*iii*) predictions using the model presented in this thesis ('new').[6] Not (anymore) surprisingly, results for the two models are *qualitatively* different. According to the traditional model, packet relaying always provides a gain, since the energy consumption of the 2-hops case is always smaller than that of the 1-hop case. In contrast, according to the actual measurements and our model, we only gain from using the relay when packets are sufficiently long (i.e., when the airtime cost becomes dominant over the cross-factor penalty).

Multicasting in WLAN

In order to multicast a packet stream from an AP to N stations in a WLAN, the following alternatives are possible: (*i*) an application layer multicast (ALM) service [120], and (*ii*) the Direct Multicast Service (DMS), part of the 802.11aa standard [121]. In the first case, the application generates a different frame for each destination; in the second case, the MAC layer takes care of replicating the frame for each station subscribed to the multicast group.

Both approaches generate the same traffic over the air. Thus, according to the traditional model they should consume the same energy, whereas we expect DMS to be *significantly* more energy efficient, since less frames cross the protocol stack. Indeed, we have experimentally verified this claim by deploying both techniques in a WLAN testbed, and measuring the relevant power consumption of the AP. The experimental settings are: $MCS = 48$ Mbps, $L = 1000$ B, $\lambda^g = 200$ fps, `txpower` $= 15$ dBm and a varying number of stations N. Fig. A.6b shows that measurements match well the model predictions.[7] More interestingly, results show that DMS can save up to 25% (1.5 W) of the total power consumption (i.e., as much as 60% of the consumption over the baseline energy cost π^{id}) with respect to ALM for $N = 10$.

Data compression in multi-hop networks

In wireless multi-hop networks, data compression has been proposed to reduce the information relayed; with such techniques, an intermediary node receives several frames, compresses them into a single frame and sends it to the next hop [122, 123].

According to traditional energy models, these approaches *surely* save energy, whereas our new energy consumption insights suggest that this may not be always true. To analyze

[6] For the model, we account for a cross-factor of 0.8 mJ to forward a packet at the relay, which has been obtained by measuring the energy consumed by forwarding a packet at the routing layer.

[7] The model accounts for a cross-factor of 0.75 mJ to reach the MAC and of 0.18 mJ from the MAC to the wireless card.

this, we used a three-node testbed consisting of a source, a sink and a relay, all using $MCS = 48$ Mbps and `txpower` $= 15$ dBm. The source node generates 500-byte packets at 1200 fps and sends them to the relay. The relay runs an application that receives these packets, and *emulates* compression by forwarding 1 frame for every m frames received. Thus, our experiments do not capture the processing toll of the compression, and hence results reflect the best possible case for the performance of this scheme.

Fig. A.6c shows total power consumption results (experimental ones, as well as predictions from old and new energy model), for different values of the compression ratio m, when data is compressed (and forwarded) at the application layer. These results are compared against the case where data is not compressed at the relay node but simply forwarded towards the sink at the routing layer.

As anticipated, the old model (top curve) predicts that compression is always advantageous. However, experimental results, matched by the new model predictions (bottom curve), show that data compression does not provide any gain in terms of energy consumption, not even for compression rates as high as 10. The reason is that the energy savings resulting from the data compression are outweighted by the extra cost of handling the packets at the application layer (cross-factor of 0.93 mJ for sending and 0.93 mJ for receiving) instead of the routing layer (cross-factor of 0.8 mJ for forwarding). This example thus shows that mechanisms devised on the basis of traditional energy models may not only fail to provide the expected gains, but may even *worsen* the actual energy consumption.

A.6.2 Novel ways to tackle energy efficiency

Packet Batching

As emerged in our work, energy consumption across the protocol stack relates to the handling of frame *units*, and is practically independent of the frame size. This suggests a straightforward energy saving strategy: *batch packets* into bundles at the highest suitable layer for a considered scenario, deliver the bundle across the stack, thus paying the energy price associated to a *single* unit, and then restore the original frames as late as possible down the stack. Unlike previous aggregation schemes for wireless networks, this mechanism (i) does not change the packets that are actually sent, but only modifies the way they are handled *within* the device [124], and (ii) does not save energy by reducing the cumulative *tail energy* consumed as a result of lingering in high power states after completing a transmission [125, 126].

We quantified the attainable energy savings by implementing the scheme depicted in Fig. A.7, which consists of (i) an "aggregator" at the application layer, which waits for n

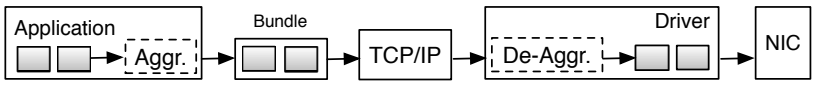


Figure A.7: Packet batching with $n = 2$.

packets to generate a bundle and pass it to the TCP/IP stack, and (*ii*) a "de-aggregator" at the wireless driver, which splits the bundle back into the original frames. Experimental measurements are reported for 100-byte packets, bundled up to an "aggregation factor" $n = 10$, and for various (application layer) frame generation rates λ^g. Frames are transmitted over the wireless channel at $MCS = 48$ Mbps and `txpower` = 15 dBm.

Results, shown in Fig. A.8, have a twofold implication. First, they provide further evidence that the cross-factor toll is practically independent of the frame size: the model matches well the measurements, and the use of an n-bundle reduces the energy toll above the driver by n. Second, energy savings are notable: with 1000 fps, an aggregation factor of 10 yields a saving of almost 0.8 W, and even the aggregation of just two packets may yield considerable savings (e.g., from 4.5 W to 4.15 W).

Obviously, casting the above described scheme into target applications (or even more general frameworks) is not straightforward,[8] and is out of the scope of this thesis. Nevertheless, the above results suggest that such effort may be rewarded with notable energy saving.

Raw sockets

Since energy is consumed while crossing each layer of the protocol stack, another way to reduce energy consumption is to *skip layers* when they are not strictly necessary (e.g., in direct host-to-host wireless communication). To quantify the relevant gains, we have implemented an application that uses `raw` sockets, thus skipping the TCP/IP OS stack. Table A.2 compares the power consumed using raw sockets ('`raw`') versus that consumed by using standard sockets ('UDP'), and reports the difference ('Δ'), for different configurations of L, MCS and λ^g (`txpower` = 15 dBm).

Results show that the cross-factor can be reduced by approx. 0.3 mJ when skipping the TCP/IP layer, in line with the results of Fig. A.2. This suggests to application developers with severe energy concerns that an extra development effort to avoid an unnecessary protocol stack may pay off.

[8]Further technical problems must be dealt with, including the interaction with the TCP/IP protocol stack (e.g., if the target application requires data to be delivered as independent TCP/IP packets) and the application's requirements (e.g., the target application scenario must tolerate the extra batching delay introduced).

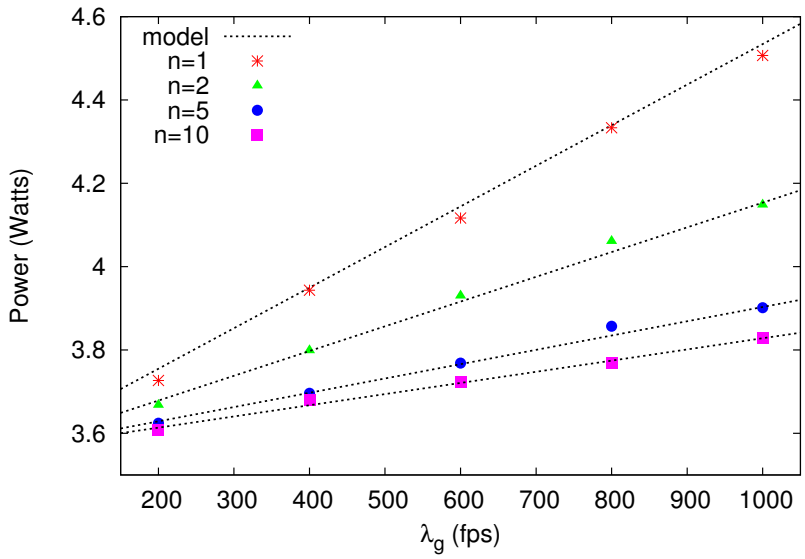


Figure A.8: Energy consumption for different 'aggregation factor' values.

Table A.2: Impact of using raw sockets.

MCS	L (bytes)	λ^g (fps)	Power (W) `raw`	Power (W) UDP	Power (W) Δ	Δ/λ^g
6 Mbps	1000	0.5k	4.45	4.60	0.14	0.28 mJ
12 Mbps	500	1k	4.50	4.79	0.29	0.28 mJ
24 Mbps	100	2k	5.05	5.54	0.46	0.25 mJ
48 Mbps	100	2k	5.03	5.52	0.44	0.25 mJ

A.6.3 Other devices

In the previous subsections, we have shown that, for the Soekris device, many existing approaches may not deliver the expected gains, while substantial savings may be achieved with novel approaches that reduce the energy consumed by packet processing in the protocol stack. To gain insight into the generality of such results, we repeated the experiments of the above sections for all the devices under study.

To evaluate the performance of the existing and novel approaches of Sections A.6.1 and A.6.2, we consider the following scenarios: (*i*) for all approaches, packets have a 100-byte payload, which is the case, e.g., of the Skype application using the preferred codec [127]; (*ii*) for the 'multicasting' experiment, we set $MCS = 48$ Mbps and $N = 10$ receivers; (*iii*) for the 'data compression' experiment, we use a compression factor of $m = 10$; and (*iv*) for the 'packet batching', we set the aggregation factor n equal to 6 packets.

Table A.3 shows (for all the devices) the energy gains delivered by the three approaches of Section A.6.1, where the gain is the percentage of per-frame energy saved by the corresponding approach, i.e., $(P_{std} - P_{app})/P_{std}$, where P_{app} is the per-frame power consumption with the approach and P_{std} is the consumption with the standard stack. In

Table A.3: Gains of existing approaches with the 'new' and 'old' models for all devices.

scheme	Soekris Linux		Soekris OpenBSD		Linksys Linksys		Alix Alix		HTC Legend		Galaxy Note 10.1		Raspberry Pi	
	old	new	old	new	old	new	old	new	old	new	old	new	old	new
relay	28.4%	-34.3%	28.4%	-36.2%	62.6%	-17.5%	67.6%	4.9%	60.6%	39.3%	68.1%	12.0%	71.8%	13.2%
multicast	0%	68.8%	0%	69.6%	0%	67.6%	0%	64.5%	0%	36.2%	0%	57.4%	0%	62.5%
compress	47.2%	-20.2%	47.2%	-21.7%	54.0%	-14.8%	45.6%	-8.2%	60.9%	29.4%	70.8%	-5.8 %	74.1%	2.1%

Table A.4: Gains of novel approaches for all devices.

scheme	Soekris Linux	Soekris OpenBSD	Linksys	Alix	HTC Legend	Galaxy Note 10.1	Raspberry Pi
batching	79.36%	80.38%	77.85%	73.42%	40.97%	66.2%	72.24%
raw	32.39%	32.81%	31.84%	29.95%	16.71%	27.01%	29.47%

particular, the table provides: (*i*) the gains predicted by the classical energy model in terms of per-frame energy cost ("old"); and (*ii*) the gains measured from our experiments, which coincide with the new energy model proposed in this thesis ("new").

We observe from the results of Table A.3 that, as with the Soekris device, the actual measurements show a very different behavior from that predicted with the classical model. With the packet relaying approach ('relay'), for some of the devices the difference with the classical model is not so drastic as to yield a performance loss, but still the gain is negligible and/or very far from the classical model. Similar conclusions hold for the data compression approach ('compress'). For the multicasting approach ('multicast'), the classical model does not predict any gain from replicating packets at the MAC layer, while measurements show very substantial gains (energy consumption is reduced by more than 35% for all devices).

Table A.4 further shows measurements of the performance gains obtained with the novel approaches of Section A.6.2. We observe that these gains are substantial for all devices: for the packet batching approach ('batching'), gains range from 40% (for the HTC Legend) to 80% (for the Soekris), while for the raw sockets approach ('raw'), we obtain more moderate but still very significant gains (in the range of 16% to 32%).

From the above results, we conclude that while the new energy model derived in this thesis has a more drastic impact on those devices with a larger *cross-factor*, it has very strong implications for all the devices under study, since in all cases (*i*) the gains of existing approaches are very different from those predicted by the classical model, and (*ii*) new approaches lead to very substantial savings.